

**CREATING NODES AND SEQUENCES.  
REPETITION, APPLYING  
TEMPLATES, MODES**

**XSLT**



# **THE <XSL:STYLESTHEET> ELEMENT AND ITS ATTRIBUTES**

## THE <XSL:STYLESTHEET> ELEMENT AND ITS ATTRIBUTES

The xsl:stylesheet element is always the top-level element of an XSLT stylesheet. The name xsl:transform may be used as a synonym.

Available in XSLT 1.0 and later versions. Available in all Saxon editions.

- Content: ( declarations )





# **ATTRIBUTES**

Type	Description
id?	<p>id</p> <p>Used to reference stylesheet modules embedded in a document.</p>
version	<p>decimal</p> <p>Standard attribute that may appear on any XSLT element. Indicates the version of XSLT required by the stylesheet.</p> <p>To request XSLT 3.0 processing (which requires Saxon-PE or Saxon-EE), setting <code>version="3.0"</code> on the <code>xsl:stylesheet</code> element is recommended, but is not sufficient on its own. An explicit request must be made on the command line or by the API. Using the XSLT 3.0 processor, a value greater than 3.0 invokes forwards compatibility mode.</p> <p>When using the XSLT 2.0 processor (in particular, using Saxon-HE), a value greater than 2.0 invokes forwards compatibility mode.</p>
default-mode?	<p>eqname   "#unnamed"</p> <p>Standard attribute that may appear on any XSLT element. Defines the default value for the mode attribute of all <code>xsl:template</code> and <code>xsl:apply-templates</code> elements within its scope.</p>
default-validation?	<p>"preserve"   "strip"</p> <p>Standard attribute that may appear on any XSLT element. Defines the default value for the validation attribute of all relevant instructions appearing within its scope. The default is strip.</p>

Type	Description
input-type-annotations?	<p>"preserve"   "strip"   "unspecified"</p> <p>Used to request stripping of type annotations. The default is unspecified.</p>
default-collation?	<p>uris</p> <p>Standard attribute that may appear on any XSLT element. Specifies the default collation used by all XPath expressions appearing in attributes or text value templates within its scope (unless overridden by another default-collation attribute on a descendant); and used by certain XSLT constructs within its scope. If present it must be a whitespace-separated list of collation URIs, that use the scheme and path <a href="http://www.w3.org/2013/collation/UCA">http://www.w3.org/2013/collation/UCA</a> (to request use of the Unicode Collation Algorithm), or collations otherwise recognized by Saxon (see Collation).</p>
extension-element-prefixes?	<p>prefixes</p> <p>Standard attribute that may appear on any XSLT element. Used to declare the use of extension instructions in a particular namespace.</p>

Type	Description
exclude-result-prefixes?	Standard attribute that may appear on any XSLT element. Used to designate namespaces as excluded. prefixes
expand-text?	Standard attribute that may appear on any XSLT element. New in XSLT 3.0, and Saxon 9.5. If set to yes, enables the use of text value templates - expressions enclosed in curly braces within text nodes, behaving the same way as attribute value templates in attribute nodes - for descendant text nodes (unless overridden by another expand-text attribute on a descendant). The default is no. boolean
use-when?	Standard attribute that may appear on any XSLT element. Used to conditionally include or exclude elements. The value is an XPath expression that can be evaluated statically. If the effective boolean value is false, then the element and all its descendants are effectively excluded from the stylesheet module. expression

Type	Description
xpath-default-namespace?	<p>uri</p> <p>Standard attribute that may appear on any XSLT element. Determines the namespace used for any unprefixed element name or type name within an XPath expression. The value may be overridden by another xpath-default-namespace attribute on a descendant. The default is a zero-length string, for names in no namespace.</p>
saxon:explain?	<p>boolean</p> <p>Saxon attribute that may appear on any XSLT element. If the value is yes, then at compile time Saxon outputs (to the standard error output) a representation of the optimized expression tree for the template or function containing that instruction.</p>

# **XSLT GLOBAL LEVEL ELEMENTS: DECLARATIONS**

## **XSLT <xsl:variable>**

The `<xsl:variable>` element is used to declare a local or global variable. Note: The variable is global if it's declared as a top-level element, and local if it's declared within a template.

Note: Once you have set a variable's value, you cannot change or modify that value!

```
<xsl:variable name="name" select="expression">  
    <!-- Content:template -->  
</xsl:variable>
```

# **CONDITIONAL COMPILATION**

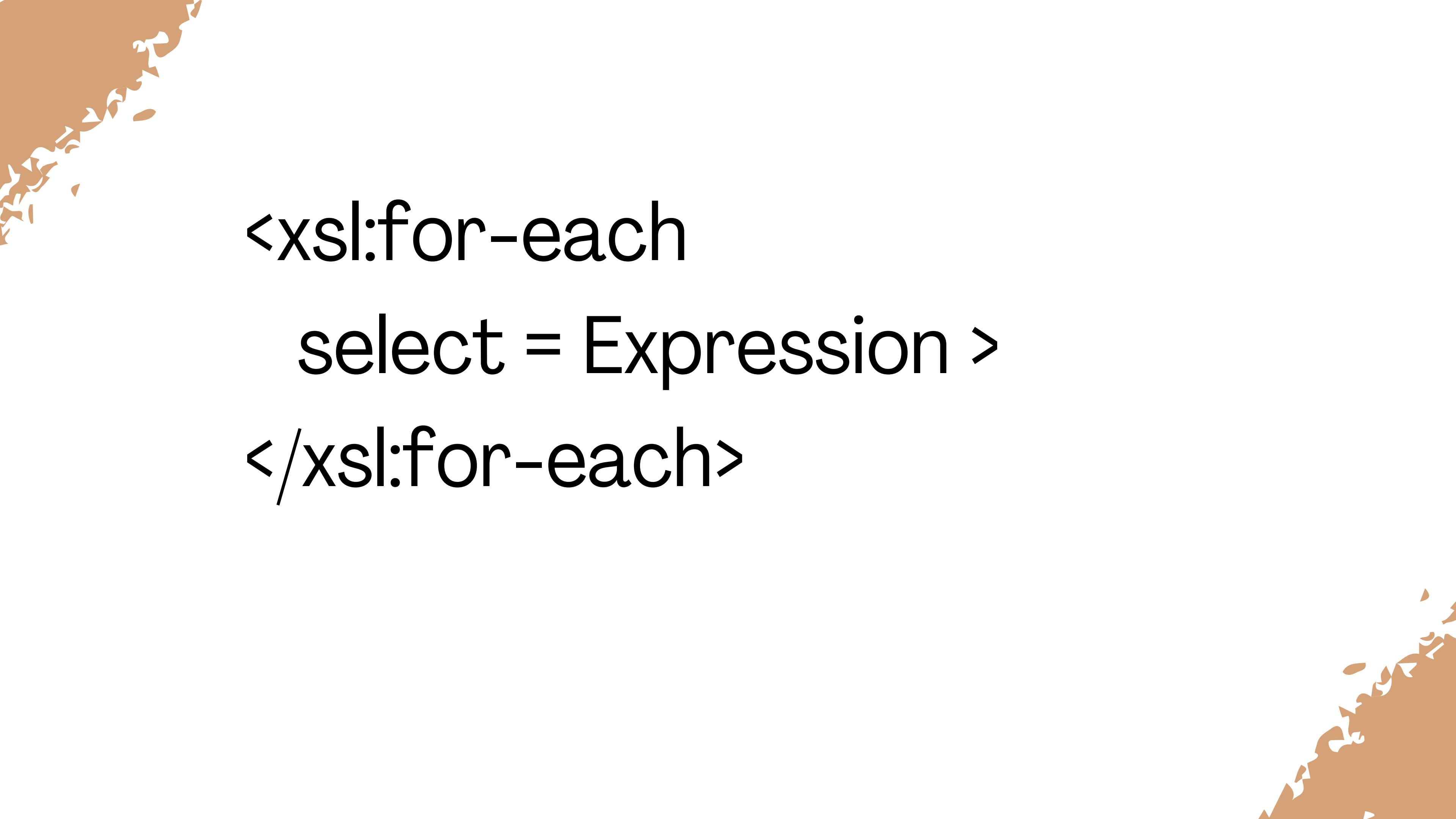
Used for conditional processing. It takes a mandatory test attribute, whose value is a boolean expression. The contents of the xsl:if element are expanded only if the expression is true.

## Example:

```
<xsl:if test="@preface">  
    <a href="preface.html">Preface</a>  
</xsl:if>
```

# xsl:for-each

`<xsl:for-each>` tag applies a template repeatedly for each node.



```
<xsl:for-each  
    select = Expression >  
  
</xsl:for-each>
```

# **<xsl:template>**

The `<xsl:template>` element defines an output producing template. This element must have either the `match` attribute or the `name` attribute set.

```
<xsl:template  
match= PATTERN  
name= NAME  
mode= NAME  
priority= NUMBER>  
<xsl:param> [optional]  
TEMPLATE  
</xsl:template>
```



# THANKS FOR WATCHING

**Thynk Unlimited**

Presentations are communication tools  
that can be used as demonstrations.