The <?xml-stylesheet?> Instruction

#?xml-stylesheet?# for a site template. #?xml-stylesheet?# description, syntax, usage, attributes and examples. HTML-5.com is a great guide for web developers. TV Series & Actors and Actresses. Follow TV Series and HTML 5 on Google+.

HTML-5.com ➢ itemscopehttp://data-vocabulary.org/Breadcrumb<span itemprop="title">HTML 5</span> ➢ itemscopehttp://data-vocabulary.org/Breadcrumb<span itemprop="title">HTML Tags</span> ➢ itemscopehttp://data-vocabulary.org/Breadcrumb

The <?xml-stylesheet?> Instruction

➢ itemscopehttp://data-vocabulary.org/Breadcrumb<span itemprop="title">Examples</span>

The <?xml-stylesheet?> Instruction for HTML 5

The `<?xml-stylesheet?>` processing instruction can be used to reference <abbr>XSLT</abbr> style sheets with templates containing HTML code and transformation instructions.

Creating a style sheet for a web site has a number of advantages:

- Site-wide style sheets with logos and navigation can be updated to easily change the <dfn>look and feel</dfn> of the entire web site or used to create multiple sites each with a different appearance from the same source of content.
- Moving repetitive code from web pages into style sheets makes the actual content of those pages more prominent, which helps improve search engine optimization.
- Different style sheets can be used for different devices, making it easier to repurpose the same content for mobile devices in addition to desktop-based browsers.
- Style sheet files can be cached on the client side by web browsers and reused as new pages are loaded without having to download them again for each page.
- The W3C Fix for Internet Explorer, which allows sending year 2000 and later HTML code to Internet Explorer (IE 6, IE 7 and IE 8) consists primarily of adding an `xml-stylesheet` instruction to the HTML document.
- Some code needed for Internet Explorer, such as the `X-UA-Compatible` meta tag, causes HTML validation errors. The code can be put into a style sheet to avoid those validation errors.

Templates in a style sheet can serve various purposes:

- templates can be used for navigation areas, possibly using HTML 5 menus, which put the same links on every page and provides a way to easily change navigation links across the entire site
- templates can be used for other common elements rather than including them on every page, which improves page load times
- templates can be used to add sections, such as a header or footer, with content that can easily be updated in one place by simply changing it in the template

back to top

<?xml-stylesheet?> Instruction Syntax

```
<a mode="pre" href="../xml-declaration/index.html#syntax"><?xml version="1.0" encoding="UTF-8"?></a>
  <b mode="pre"><?xml-stylesheet ...?></b>
  <a mode="pre" href="../doctype-declaration/index.html#syntax"><!DOCTYPE html></a>
  <a mode="pre" href="../html-tag/index.html#syntax"><html xmlns="http://www.w3.org/1999/xhtml"></a>
     ...
  </html>
```

Rules for coding the xml-stylesheet instruction for HTML 5

Make sure you understand the difference between a tag and element and are familiar with the definitions of namespace and other HTML terms.

- The `xml-stylesheet` processing instruction is optional. Include a `xml-stylesheet` instruction with the same href attribute on all web pages that use the same set of web site templates.
- It should be coded between the xml declaration and the DOCTYPE declaration.
- The processing instruction starts with the characters `<?`.
- The tag name is `xml-stylesheet`. The name is in the HTML namespace, even if the namespace is not explicitly specified in the <html> tag.
- The value of the type attribute should be `text/xsl`.
- The href attribute should be a URL that references a valid style sheet.
- The processing instruction is terminated by the `?>` characters.
- Processing instructions are self-closing tags and therefore do not have a corresponding end tag.

Style Sheet and Templates

```
  <xsl:stylesheet version="1.0"
     xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
     xmlns:html="http://www.w3.org/1999/xhtml"
     xmlns="http://www.w3.org/1999/xhtml"
     exclude-result-prefixes="xsl html"
  >
     <xsl:output method="html" version="1.0"
        omit-xml-declaration="no"
        doctype-system="about:legacy-compat"
     />

  <!-- This is the "Fix for IE" based on the example in W3C's FAQs:
     http://www.w3.org/MarkUp/2004/xhtml-faq#ie
  -->

     <xsl:template match="/html:html/html:body">
        <xsl:copy>
           <xsl:apply-templates select="@*"/>
           <p><var mode="pre"><#Template header code goes here.#></var>  </p>
           <xsl:apply-templates select="node()"/>
           <p><var mode="pre"><#Template footer code goes here.#></var>  </p>
        </xsl:copy>
     </xsl:template>

     <xsl:template match="@*|node()">
        <xsl:copy>
           <xsl:apply-templates select="@*|node()"/>
        </xsl:copy>
     </xsl:template>
```

```
</xsl:stylesheet>
```

Rules for coding the style sheet and site templates

- The main element of the stylesheet document is a **stylesheet** element. Code a starting `<xsl:stylesheet>` tag at the beginning of the document and an ending `</xsl:stylesheet>` tag at the end of the document. The tag's element type name, `stylesheet`, uses lower case letters and should be in the XSL namespace which will be associated with the namespace prefix before the element name (`xsl` in our examples).
- On the starting <stylesheet> tag, code the `version` attribute, which is required and indicates the version of XSLT that it to be used to interpret the style sheet and transform HTML documents.
- Indicate the namespaces used in the style sheet and templates by coding a xmlns attribute with a prefix for each namespace and one without a prefix for the default HTML namespace. (For backward compatibility, the namespace from the 1999 W3C standard is used for all HTML versions from that year forward, including XHTML and HTML 5.) The style sheet will include at least two of the possible namespaces in HTML 5:
1. `http://www.w3.org/1999/XSL/Transform` to reference the XSLT elements
2. `http://www.w3.org/1999/xhtml` to match HTML tags and generate HTML code
- Include an `exclude-result-prefixes="xsl html"` attribute in the starting <xsl:stylesheet> tag so that the only xmlns attribute on the <html> tag in the result document is the xmlns attribute without a prefix for the default HTML namespace.
- Immediately after the starting `<stylesheet>` tag, code an **output** element. The tag's element type name, `output`, uses lower case letters and should be in the XSL namespace which will be associated with the namespace prefix (`xsl`) before the element name. Include a `method="html"` attribute on the element. Note: without the `<output>` tag and `method="html"` attribute, the DOM will be inconsistent between browsers. Specifically, in some browsers, such as Opera, the `document` object will be an HTMLDocument while in other browsers, such as Firefox, the `document` object will be a XMLDocument. Since some methods, such as getElementsByName are not supported on an XMLDocument object, certain JavaScript code will not work correctly when the **output** element or the `method="html"` attribute is omitted.
- Inside the **stylesheet** element, between the starting `<stylesheet>` tag and ending `</stylesheet>` tag, code the templates for the web site. For example, a template with `match="/html:html/html:body"` can be included to add a header and footer to the HTML body section.
- Include a template with an identity transform:

```
    <xsl:template match="@*|node()">
       <xsl:copy>
          <xsl:apply-templates select="@*|node()"/>
       </xsl:copy>
    </xsl:template>
```

back to top

<?xml-stylesheet?> Instruction Attributes

common attributes	In addition to the attributes below, which are specific to the <a> tag, any of the common HTML attributes can also be coded
`type`	`type="text/xsl"`
`href`	

back to top

<?xml-stylesheet?> Instruction Examples

Examples of the `<?xml-stylesheet?>` instruction in HTML 5

Include an `<?xml-stylesheet?>` processing instruction between the xml declaration and the DOCTYPE declaration. The `xml-stylesheet` is included for older browsers. It serves the same purpose as the HTML <link> tag with `type="application/xslt+xml"` for HTML version 5, which should also be included:

<samp>index.html</samp>

```
<?<a mode="pre" href="../xml-declaration/">xml</a>   version="1.0" encoding="UTF-8"?>
<b mode="pre"><?xml-stylesheet type="text/xsl" href="/site-template.xsl"?></b>
<a mode="pre" href="../doctype-declaration/"><!DOCTYPE html></a>
<a mode="pre" href="../html-tag/index.html#" title="<html> tag"><html xmlns="http://www.w3.org/1999/
   <a mode="pre" href="../head-tag/"><head></a>
      <<a mode="pre" href="../link-tag/">link</a>   rel="stylesheet" type="application/xslt+xml" hre
      <<a mode="pre" href="../title-tag/">title</a>  >Example Only</title>
   </head>
   <a mode="pre" href="../body-tag/"><body></a>
      <a mode="pre" href="../heading-tags/"><h1></a>  Here Is Your Heading</h1>
      <a mode="pre" href="../p-tag/"><p></a>  This is the first paragraph of your web page.
      </p>
   </body>
</html>
```

<samp>site-template.xsl</samp>

```
<xsl:stylesheet version="1.0"
   xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
   xmlns:html="http://www.w3.org/1999/xhtml"
   xmlns="http://www.w3.org/1999/xhtml"
   exclude-result-prefixes="xsl html"
>
   <xsl:output method="html" version="1.0"
      omit-xml-declaration="no"
      doctype-system="about:legacy-compat"
   />

<!-- This is the "Fix for IE" based on the example in W3C's FAQs:
   http://www.w3.org/MarkUp/2004/xhtml-faq#ie
-->

   <xsl:template match="/html:html/html:head">
      <xsl:copy>
         <xsl:apply-templates select="@*"/>
         <!-- Workaround for Microsoft Bug #s 636728 and 572123
            X-UA-Compatible meta is ignored when inside conditional comments
         -->
         <meta http-equiv="X-UA-Compatible" content="IE=9"/>
         <xsl:apply-templates select="node()"/>
      </xsl:copy>
   </xsl:template>
   <xsl:template match="/html:html/html:body">
      <xsl:copy>
         <xsl:apply-templates select="@*"/>
         <p>This header comes from a template in the style sheet.</p>
         <xsl:apply-templates select="node()"/>
         <p>This footer also comes from the style sheet template.</p>
      </xsl:copy>
   </xsl:template>

   <xsl:template match="@*|node()">
      <xsl:copy>
         <xsl:apply-templates select="@*|node()"/>
      </xsl:copy>
   </xsl:template>

</xsl:stylesheet>
```

back to top

Changes in HTML 5 - <?xml-stylesheet?> Instruction

What's new in HTML 5

Differences between HTML 5 and earlier versions of HTML

In HTML 5, a `type="application/xsl"` style sheet can be specified in an HTML link tag. Use of the `<?xml-stylesheet?>` processing instruction in addition to the link tag is recommended for backward compatibility with older browsers.

back to top

THE END