

Style vs. Semantics - Intro to CSS

HTML-5.com is an HTML User's Guide and quick reference of HTML elements and attributes for web developers who code HTML web pages, not only for HTML 5 but for HTML coding in general, with demos and examples of HTML code plus a cheat sheet for web developers. [TV Series & Actors and Actresses](#). Follow [TV Series](#) and [HTML 5](#) on Google+.

[HTML-5.com](#) > [itemscopehttp://data-vocabulary.org/Breadcrumb](http://data-vocabulary.org/Breadcrumb)HTML 5 > [itemscopehttp://data-vocabulary.org/Breadcrumb](http://data-vocabulary.org/Breadcrumb)HTML 5 [HTML Tutorials](#) > [itemscopehttp://data-vocabulary.org/Breadcrumb](http://data-vocabulary.org/Breadcrumb)HTML Tutorials [Style vs. Semantics - Intro to CSS](#)

Separation of Style from Content and Structure

migration toward separation of style from content and structure

Previous versions of HTML have deprecated presentational elements that provide no semantic value and are used exclusively to style text. HTML 5 continues the move in this direction, encouraging styles to be applied through use of [CSS](#).

separation of content and layout

One of the most significant improvements of the 2000-2001 version of HTML over the 1997-1999 version was the ability to take advantage of [XSLT](#) style sheets. Using templates allows separating the content of individual pages from the page layout and general structure of related pages. For example, you could create templates for the layout of various sections of a web site, such as a shopping cart, as well as import a site-wide template for the entire site's "look and feel". Templates are stored in separate documents, which web browsers only need to cache once, thus improving page load times. HTML version 5 continues to support these templates, which were not available in HTML version 4.

Both XSLT and CSS style sheets for any number of HTML pages can be referenced by the [link](#) tag which allows creating specific layouts for various devices. For example, the template and styles for a printer-friendly version would be indicated by [link media="print" .../>](#) while a version for small screen devices such as cell phones would be indicated by [link media="handheld" .../>](#).

case-sensitive tags and attributes

In the 2000 W3C recommendation and later versions of HTML, HTML tag names should be coded in all lower case. Attributes on HTML elements and CSS selectors in a [class attribute](#) are also case-sensitive.

consistent namespace for HTML elements (either explicit or implied)

In HTML version 4, the tag element names were unnamespaced. If you applied a style sheet (such as a site-wide template for the site's "look and feel") to a "well-formed" HTML 4 document (where all tags were properly closed with matching end tags) the element names would have to be matched without a namespace prefix (for example: [img](#)). The template would use [xsl:output method="html" omit-xml-declaration="yes"/>](#) to convert the document to true HTML, by removing extraneous closing tags, for example. That is, coding in the source document such

as `` would automatically become `` with no closing `` tag or self-closing `/>`.

In the 2000 W3C standard, a namespace is explicitly specified, usually within the top element, which applies to all HTML elements (`<html xmlns="http://www.w3.org/1999/xhtml">`). Therefore, template matching rules would include a prefix for the namespace (ex: `match="html:img" xmlns:html="http://www.w3.org/1999/xhtml"`).

In HTML version 5, the namespace can be explicitly specified in well-formed HTML documents or, in HTML syntax documents, the "http://www.w3.org/1999/xhtml" namespace is implied, which allows the matching rules with namespace prefixes to be applied to either syntax.

[back to top](#)

THE END