Find Out Which Browsers are Compatible with Various CSS Features

Use <https://quirksmode.org/css/>.

Alphabetized List of CSS Features

Use <https://www.w3.org/TR/CSS21/propidx.html>

CSS Selectors

There are 3 ways to express selectors.

1. body { background-color: #cccc99; } “simple” selector - applies the style to HTML elements named “<body>”.
2. #menu { background-color: #cccc99; } “ID” selector - applies the style to HTML elements whose ID is “menu”.
3. .bookTitle { background-color: #cccc99; } “class” selector - applies the style to HTML elements whose class name is “bookTitle”.

Stacking CSS Selectors

h1 { background-color: #cccc99; } – applies the background color to all <h1> elements

whereas

h1,h2 { background-color: #cccc99; } – applies the background color to all <h1> and <h2> elements.

CSS Selectors via More Complex Patterns

|  |  |
| --- | --- |
| /\* descendent selector \*/  div p { background-color: # ddddaa; }  /\* child selector (notice the > between “div” and “p”) \*/  div > p { background-color: # cccc99; } | <div>  <form>  <p>I’m a descendent, but not a  child.</p>  </form>  <p>I’m a child.</p>  </div> |

The background color of the 1st paragraph is RGB:ddddaa, because this paragraph is a descendent. The background color of the 2nd paragraph is RGB:cccc99, because it is child – an immediate descendent. Yes, this paragraph is also a descendent, but the cascading rules apply - the selector rule placed later overrides an earlier rule that also applies.

|  |  |
| --- | --- |
| /\* attribute selector \*/  img[alt=spacer] { padding:0x;} | <img src=“gradient.jpg” alt=”spacer” /> |

The “padding” style applies to any image element that sets its “alt” attribute to “spacer”.

|  |
| --- |
| /\* pseudo class \*/ a:visited {color: #dddddd; } |

Set the color of an anchor tags if the user has already visited the URL expressed in the anchor tag.

Size expressed in “em”s

“Ems” (em): The “em” is a scalable unit that is used in web document media. An em refers to the current font-size; for instance, if the font-size of the document is **12pt**, **1em** is equal to **12pt**. Ems are scalable in nature, so **2em** would equal **24pt**, 0.5**em** would equal **6pt**, etc.

One stylesheet referring to another stylesheet

The statement

@import url(“styles2.css”);

can be placed inside a stylesheet; it instructs the browser to open a 2nd stylesheet whose URL is style2.css, and apply its rules at the position of the @import statement.

Font Collections

|  |  |
| --- | --- |
| Family Name | Collection Name |
| Times New Roman | serif |
| Arial | sans-serif |
| Comic Sans | cursive |
| Impact | fantasy |
| Courier New | monospace |

“monospace” means “the same amount of space for each letter”. Font size is commonly expressed via the font family, not the specific font. There are a few situations in which we will want to specify a given font; in that case it is good practice to list the fonts after “font-family:” via a comma-separated list with the 1st preference, followed by the 2nd preference, . . . , and a font family listed last (as a default). For example

font-family:'Times New Roman', serif;

Preformatted Text

Use the <html> selector “<pre>” for preformatted text. The browser is instructed not to change spacing nor line breaks.

Flavors of Positioning

The style rule **position:** applies to any HTML element for which the concept “box” applies (refer to “CSS and the Box Model”); examples of these are <div>, <h1>, <p>, <article>, etc. As an alternative to the term “box” model, the word “block” also applies to these HTML elements.

There are 4 fundamental flavors for the **position:** rule: S*tatic, Relative, Absolute,* and *Fixed*.

S*tatic* positioning is the default. It instructs the browser to place this HTML element immediately below the previous block element.

*Relative* positioning takes offset values – d pixels down from the top edge of the box had this been *Static*, and r pixels to the right of the left edge of the box had this been *Static*.

*Absolute* and *Fixed* are similar. Both of these – like *Relative* positioning - take offset values. But the offset values for *Absolute* are relative to the upper-left corner of <body>’s box. And the offset values for *Fixed* are relative to the upper-left corner of the window in which the HTML document is displayed. *Absolute* and *Fixed* positioning behave the same until the document becomes larger than the window in which it resides; when the document overgrows its window boundaries the user can scroll the document (left, right, up, or down). A block with *Absolute* positioning will move when the user scrolls the document; a block with *Fixed* positioning will remain in place when the user scrolls the document.