CSS in HTML

External CSS

<link rel="stylesheet" href="/styles/screen.css" type="text/css" media="screen"%3E

Embedded CSS

<style type="text/css" media="screen"> /*Style declarations here*/</style>

Good practice requires providing styles for screen and print media as a bare minimum. Multiple media types should be specified as a comma separated list. Other media types include all, aural, braille, embodied, handheld, print, projection and tv.

Inline CSS

<tag style="..."> where tag is any HTML element tag such as h1, p, a etc. Use single quotes in the style assignment if required. Follow the property1:value1, format described above. Use sparingly.

Length Units

Absolute Units: in|cm|mm|pt|pc

px (Pixels) as defined is a relative unit but as used by browsers is a universal unit.

Relative Units: em% Measurements are relative to the parent element.

Lengths, except when 0, must always be followed by a unit. No space allowed between the number & the unit.

Color Units & Names

#RRGGBB is, generally speaking, the most economical way of specifying color. e.g. "#FF0000"

Color Names: aqua|black|blue|fuchsia|green|lime|maroon|navy|olive|purple|red|silver|teal|white|yellow

Background[1]

background-attachment: fixed|inherit
background-color: transparent|<color>|inherit
background-image: none|url(<urlname>)

background-position: left|center|right|top|bottom|left|top|right|bottom
L is an offset from top-left. Default 0 0 background-repeat:no-repeat|repeat|repeat-x|repeat-y

Border[23]

border-color: transparent|<color>|inherit
border-style: none|hidden|dotted|dashed|solid
double|groove|ridge|inset|outset
border-width: medium|thin|thick|<length>|inherit

? is one of left, top, right or bottom

Color

color:inherited

Default depends on the browser. Also sets border color unless overridden.

Cursor

cursor:auto|default|pointer|crosshair|move|pointer-down| resize|text|text|url|help|progress|inherit

where = no|new|new|new|sw

Use to assign cursor shape when mouse pointer enters the HDE. What "enter" means is browser dependent.

Dimensions

Dimension specifications only make sense for

● Positioned HDEs, i.e. HDEs which have position set to a value other than static
● floated HDEs, i.e. HDEs which have float set to left or right, in which case a width must be specified.

Bottom/top/lef/right: auto|<length>|inherit

top & bottom and left & right form pairs. If both are set to auto, both are zero. If one is set to auto, it is the negative of the other. If both are assigned a length and one is not the negative of the other, the browser may, at its discretion, ignore the setting.

width|height: auto|<length>|inherit

width & height are only relevant for block HDEs.

max|min?:<length>|inherit where ? = width|height. The max properties are 0 by default. The min properties have no initial value. Excess content spills out of the block by default. Assign overflow to avoid this. IE5

Display

display:none|inline|inline-block

Determines the type of display be generated by an HDE during page layout. The default value is HDE dependent <span>and <div> HDEs for instance are inline while <div> and <p> HDEs are blocks. Floating an HDE sets display:block.

Inline HDEs only take up as much horizontal space as required for displaying their contents. Block HDEs take the available client area of their container – unless width or max-width require otherwise.

inline-block causes the HDE to behave like an inline element while still using all block like attributes such as margin and padding. For compatibility with Firefox use display:-moz-inline-stack;display:inline-block

display:none hides the HDE and generates no display block for it. Other HDEs are laid out as though they were not there.

Float

float:none|left|right|inherit

Causes HDE to "float" to the left or right of its container if empty or to the right/left of its previously floated sibling if non-empty. Floated HDEs are always blocks. The width property must be specified. If available client width is insufficient floating begins afresh from the appropriate boundary of the container - below previously floated siblings.

Assign clear: none|left|right|both to force subsequently floated HDEs to start afresh from the appropriate boundary of the container – a sort of "carriage return" directive.

Font

font-family: inherit|<fset>|<fset>|<fset>|<fset>

font-size: inherit|<length>

font-variant: normal|italic
font-weight: normal|bold
font-stretch: inherit|normal
font-style: inherit

FontSize — font-size: <font-size>|<font-family>|caption|icon|menu|message-box|small-caption|status-bar

List

list-style-image: none|url(<urlname>)|inherit

list-style-position: outside|inside|inherit

list-style-type: disc|circle|square|decimal|decrement-leading-zero|lower-roman|upper-roman|lower-greek|upper-greek

lower-latin|inherit

Margin

margin?:auto|<length>|inherit

? is one of left, top, right or bottom

Overflow

overflow:auto|visible|hidden|scroll|inherit

Paddin

padding?:auto|<length>|inherit

? is one of left, top, right or bottom

Position

position:static|relative|absolute|fixed|inherit

Absolutely positioned HDEs have as their container the nearest ancestor that has a position attribute other than static. If no such ancestor exists, the container is the initial containing block – for all intents the browser window.

Fixed HDEs always have the viewport as their container. Not very useful until IE6 becomes obsolete.

Absolute HDEs define their own stacking context. This means that z-index settings for their children are internal to the HDE.
Text

alignment: left|right|center| justify

text-decoration: none|underline|overline|line-through

text-indent: |L|> inherit

Text

Indent first line of block HDEs. Use negative values to outdent.

text-transform: none| uppercase| capitalize

vertical-align: baseline| sub| super| top| text-top

middle| bottom| text-bottom| > inherit

Relative <L> values refer to the line-height of the element itself. Use -ve <L> values for subscripts. +ve values are an offset from the line top while -ve values are an offset from the line bottom.

line-height: normal|<N>|> inherit

<N> = scaling factor. Use <N> < 1 or > 100% to compress lines in block HDEs. Defines minimum line-height. Maximum is determined by inline elements, such as images.

Visibility

visibility: inherit|visible| hidden| collapse

HDEs are not visible but still take up space and affect the layout of the document. collapse causes table rows/columns to be hidden.

Element Types

Replaced elements are rendered by the browser using information not available in the document itself. Examples are images, <img>, and inputs, <input>. Nearly all other elements are non-replaced, i.e. their content is available in the document itself and is displayed in a box specified by the element attributes.

Block elements generate a box that, unless otherwise specified, occupies the full client width of the parent. Examples: <div>, <p> & <i>. The vertical margins of adjacent blocks collapse – i.e. only the bigger of the two is used. This does not apply to elements that are blocks by virtue of being float.

Inline elements generate a box which only consumes as much horizontal space as is required to display its contents. Inline elements do not use vertical margin settings. Examples: <a>, <span>, <b> etc.

Block elements can act as containers for other elements

Element types can be changed by setting the display attribute. inline-block (-moz-inline-stack) offers a useful half-way house between block and inline.

Box Model

HDEs are rendered by browsers in a notional box made up of the following:

1. Margin
   1. vertical, where relevant.
   2. horizontal, always
2. Padding
3. The border
4. The client area

The precise manner in which CSS height/width settings are used is browser and DTD dependent

With the strict DTD the figures reported by Javascript for window.innerWidth/Height include the padding but not the border. window.offsetWidth/Height, on the other hand, reports figures that include both the border width and the padding. Padding is nevertheless outside the client area – i.e. the available area for child HDEs is CSS height x CSS width.

With the transitional DTD nothing changes – with Firefox and Opera. However, IE (even IE7) treats the CSS settings as the whole area of the element – border and padding included. The padding and border eat into the client area.

Notes

Tested under IE6+, Firefox 1.5+ & Opera 9.1+.

Most CSS properties are not inherited. Notable exceptions are color, font, letter-spacing, word-spacing, text-align, text-indent & line-height.

CSS properties describe HDE qualities. In other words, they are always adjectives, not verbs. Thus visibility:hidden not visibility:hide.

Key

rule – CSS Selector
property – CSS Property Identifier
value – CSS Property Value
value – CSS Property Value (Default)
attrib – HTML Element Attribute Identifier
value – HTML Element Attribute Value

<text> - replace text with suitable value
C – color value, e.g. fuchsia or #FF00FF
F – Family-name, e.g. arial, “comic sans ms” etc. Names containing spaces should be wrapped in quotes.
G – Generic-family name = serif|sans-serif|cursive|fantasy|monospace. Optional but recommended.
L – length value with unit, e.g. 2em, 10% 30px etc
N – simple number (no unit)
– accepts shorthand. Shorthand properties are specified by combining the individual property specifications and assigning them to the root property, e.g. border: or font: Note: Omitting a sub-property in shorthand specification causes it to take its default value. With the notable exception of font:, shorthand properties can be specified in any order.
1 IE6 bug: 20% right will move background right. Firefox and Opera will use 20% and ignore right.
2 There are five distinct shorthand formats for the border property
   border-?; where ? is one of left, top, right or bottom
   border: value
3 for consistency specify a border width in length units. Each browser has its own interpretation of medium.
4 only a partial list
5 Not a copy of CSS standards. Use relative units for font sizes. Always specify font information – the defaults depend on, user-configured, browser settings. Other font-weights are often not available in the selected font. The shorthand, font, property requires the sub-property order given here to be followed. The system settings options, such as menu and caption, are handy when creating dialogs which feel like system dialogs.
6 The default is zero. Assign shorthand margin first, then change others individually - if required. Vertically adjacent block HDEs collapse – i.e. only the bigger margin is used. Inline HDEs do not use vertical margins. Horizontal margins never collapse. Nor do any of the margins of floated HDEs. To center HDE horizontally in its parent, set horizontal margins to auto. Margins can be negative – use with care for overlap effects.

An extensive range of free quick reference cards is available at http://www.explainth.at