# Blocks and Inline Elements Block-level Elements

Block elements are elements that separate a block of content.

* **Lists (<ul> and <ol>)**
* **Tables (<table>)**
* **Forms (<form>)**
* **Divs (<div>)**

# Inline Elements

Inline elements are elements that affect only snippets of content and do not block off a section of a page.

* **Links (<a>)**
* **Images (<img>)**
* **Formatting tags (<b>, <i>, <tt>, etc.)**
* **Phrase elements (<em>, <strong>, <code>, etc.)**
* **Spans (<span>)**

**Paragraphs: <p>**

**Heading Levels:** <h1>, <h2>, <h3>, <h4>, <h5> and <h6>, descending in importance from

**Breaks and Horizontal Rules:** The <br> tag forces a line break. The <hr> tag creates a horizontal rule across the page.

**Quoted Text:** The <blockquote> and <q> elements are used to designate quoted text.

**Preformatted Text:** Occasionally it is desirable to output content as it is laid out in the code, whitespace and all. The <pre> tag is used for this purpose.

**Phrase Elements:** Phrase elements provide structural information about the content they contain. The most common phrase elements are <em> and <strong>.

**Documenting Changes:**

**Text Links:**

<a href="path">Link Text</a>

**Absolute vs. Relative Paths**

<a href="/path">Home Page</a>

<a href="/about/Company.html">About Our Company</a>

**External Links**

<a href="http://www.google.com">Google</a>

**Inserting Images:**

* **src**
* **alt**
* **Height**
* **Width**

The <img> tag is an empty tag. Its src attribute is used to reference an image le using a relative or absolute path.

<img src="path " alt="alternative\_text">

**Image Links:** Simply put an <a> anchor tag around the image.

<a href="index.html" title="Link to Home Page"><img src="images/file.gif"></a>

**Unordered Lists**

The <ul> tag starts an unordered list. Each list item is contained in <li></li> tags.

**Ordered Lists**

The <ol> tag starts an ordered list. Each list item is contained in <li></li> tags.

* **Start Attribute**

**Definition Lists**

The <dl> element contains the definition list. The <dt> elements are the definition terms and the <dd> elements are the definition descriptions.

**Creating Tables**

The tag for creating tables is <table>. The table row tag is <tr> and the two tags that can create table cells are <td> and <th>.

**<table> Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| width | sets the width of the table in pixels or percentage |
| border | sets the width of the border of the table |
| cellspacing | sets the distance between cells in pixels |
| cellpadding | sets the distance between cell content and the cell border in pixels |
| summary | used to provide a longer description of the table for screenreaders |

**<tr> Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| align | aligns the content in the cells in the table row horizontally (left, center, or right) |
| valign | aligns the content in the cells in the table row vertically (top, middle, or bottom) |

**<td> and <th> Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| align | aligns the content in the cells in the table row horizontally (left, center, or right) |
| valign | aligns the content in the cells in the table row vertically (top, middle, or bottom) |
| colspan | indicates the number of columns the cell should span |
| rowspan | indicates the number of rows the cell should span |

**Creating form**

HTML forms are created using the <form> tag, which takes two main attributes: action and method.

<<add more>>

# CSS Rules

CSS *rules* are statements that define the style of an element or group of elements.

selector {

property: value;

property: value;

property: value;

}

Each *property:value* pair is a *declaration*. Multiple declarations are separated by semicolons. The *selector* defines which elements are affected by the rule.

**CSS Comments**

Comments in CSS begin with "/\*" and end with "\*/".

p {

color: red; **/\* All paragraphs should be red \*/**

}

**Selectors**

**Types of Selectors**

* **Type:** Type selectors specify elements by tag name and affect every instance of that element type.

**p** {

  color: darkgreen;

font-family: Verdana;

font-size: 10pt;

}

* **Descendant:** Descendant selectors specify elements by ancestry.

**p strong** {

color: red;

}

* **Child:** Child selectors specify a direct parent-child relationship and are indicated by placing  a > sign between the two tag names.

**p > strong** {

color: red;

}

* **Class:** The names given to classes are arbitrary, but should be descriptive of the purpose of the class.

**.warning** {

font-weight: bold; color: #ff0000;

}

<h1 **class="warning"**>WARNING</h1>   
<p **class="warning"**>Don't go there!</p>

* **ID:** The id attribute, which is used to uniquely identify an element on the page
* **Attribute:** Attribute selectors specify elements that contain a specific attribute.

**a[target]** {

color: red;

}

* **Universal:** The universal selector is an asterisk (\*). It matches every element.

**\***{

color: red;

}

* **Grouping:** Selectors can share the same declarations by separating them with commas.

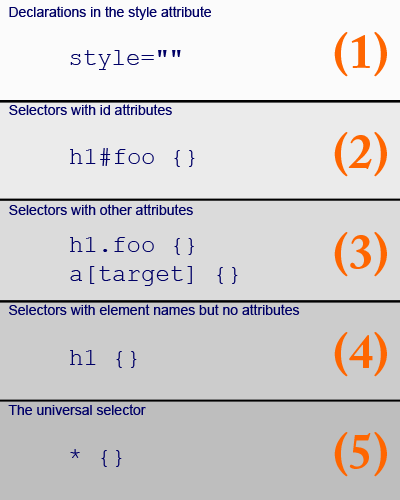
i, .warning, #important {

text-decoration: underline;

}

**Precedence of Selectors**

* The rule with the more specific selector takes precedence.
* In the event that two selectors have the same specificity, the rule specified later in the document takes precedence.



**The Cascade**

**Embedded Style Sheets**

Embedded style sheets appear in the style element in the head of an HTML page.

**External Style Sheets**

External style sheets are created in separate documents with a ".css" extension.

**Inline Styles**

Inline styles are created by adding the style attribute to a tag.

**<div> and <span>**

The <div> and <span> tags are used in conjunction with Cascading Style Sheets.

**Units of Measurement**

CSS allows you to specify font size, border size, margins, padding, etc.

|  |  |  |
| --- | --- | --- |
| **Unit** | **Description** | **Example** |
| px | Pixels | margin-top: 10px; |
| pt | Points | font-size: 12pt; |
| in | Inches | padding-top: .5in; |
| cm | Centimeters | top: 5cm; |
| mm | Millimeters | left: 45mm; |
| pc | Picas | bottom: 12pc; |
| em | Ems | font-size: 1.5em; |
| ex | Exs | font-size: 1.5ex; |
| % | Percentage | width: 80%; |
| rem | Rems | font-size: 1.7rem; |

# CSS Fonts

**Font-family**

**Specifying Font by Category**

* cursive  - Comic Sans
* fantasy  - Castellar
* monospace - Courier
* sans-serif - Arial
* serif – Time New Roman

p { font-family: Arial, Helvetica, sans-serif; }

If neither Arial nor Helvetica is found, the browser at least knows to use some sans-serif font.

<https://www.thoughtco.com/web-safe-fonts-3467430>

**Font-size**

**Relative Font-size Terms**

In addition, font size can be de ned using the following relative terms:

**•** xx-large (24pt) **•** x-large (17pt) **•** large  (13.5pt) **•** medium (12pt) **•** small (10.5pt)  **•** x-small(7.5pt)

**•** xx-small(7pt) **•** smaller **•** larger

font-size: smaller;

font-size: 500%;

font-size: smaller;

font-size: 12pt;

font-size: 2em;

**Font-style**

**•** normal **•** italic **•** oblique

**Font-variant**

**•** normal **•** small-caps

**Font-weight:** determines how thick (or bold) it is

**•** 100 **•** 200 **•** 300 **•** 400 **•** 500 **•** 600 **•** 700 **•** 800 **•** 900 **•** bold **•** bolder **•** lighter **•** normal

font-weight: 400;

Font property with various font preferences into a single rule

font-style | font-variant | font-weight | font-size | font-size | font-family

# CSS Text

**Letter-spacing**

**Word-spacing**

**Line-height**

**Text-align**

**•** left  **•** right **•** center **•** justify

**Text-decoration**

**•** none **•** underline  **•** overline  **•** line-through  **•** blink

**Text-indent:** The value can be specified in number of units or in percentage of the width of the containing block.

**Text-transform:**

**•** none **•** capitalize **•** uppercase **•** lowercase

**Text-shadow**

* h-shadow - the position of the horizontal shadow; value can be negative (required)
* v-shadow - the position of the vertical shadow; value can be negative (required)
* blur-radius - the radius of the blur (optional; default is 0)
* color - the color of the text shadow (optional)
* none - no text shadow (default value)

**Vertical-align**

**•** bottom **•** middle **•** top **•** text-bottom **•** baseline

**•** text-top **•** sub **•** super

**White-space**

# Backgrounds and Colors

https://www.w3.org/TR/css3-color/#html4

**Color**

**Color Names**

**Hexadecimal Color Values:** Hexadecimal color values take the format #rrggbb, where rr is the amount of red in the color, gg is the amount of green in the color, and bb is the amount of blue in the color.

selector {

color: red;

}

**Short Hexadecimal Color Values:** Hexadecimal color values can be abbreviated when a color is represented by three *pairs* of hexadecimal characters.

selector {

color: #f60;

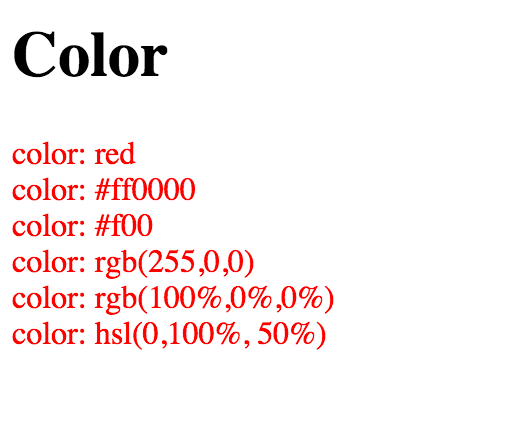
}

**Functional Notation**: Functional notation takes the format rgb(n,n,n), where n is a number between 0 and 255 or percentage between 0% and 100% for the red, green, and blue intensities of the color, respectively.

selector {

color: rgb(255,0,0);

}

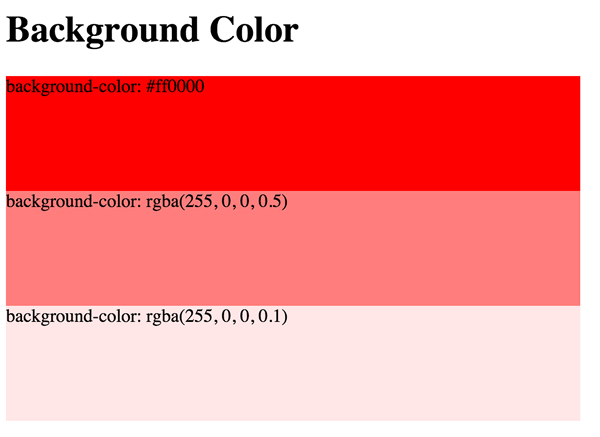


**Background-color:** used to specify the background color of an element. It can be applied to block elements and inline elements.

selector {

background-color: color;

}



**Background-image:** used to specify the background image of an element. It can be applied to block elements and inline elements.

selector {

background-image: url(images/bg.png);

}

**Background-repeat:** used with background-image to specify whether and how a background image should repeat.

* **no-repeat - does not tile**
* **repeat-x - tiles horizontally**
* **repeat-y - tiles vertically**

selector {

background-image:url(url);

background-repeat:value;

}

**Background-attachment:** used with background-image to specify whether a background image should scroll as the content is scrolled or whether the content should scroll over it.

**•** scroll **•** fixed

selector {

background-image:url(url);

background-attachment:value;

}

**Background-position**

* top  right  bottom  left
* center  any combination of the above

selector {

background-image:url(url);

background-position:value;

}

**Background-size**

**CSS and Links**