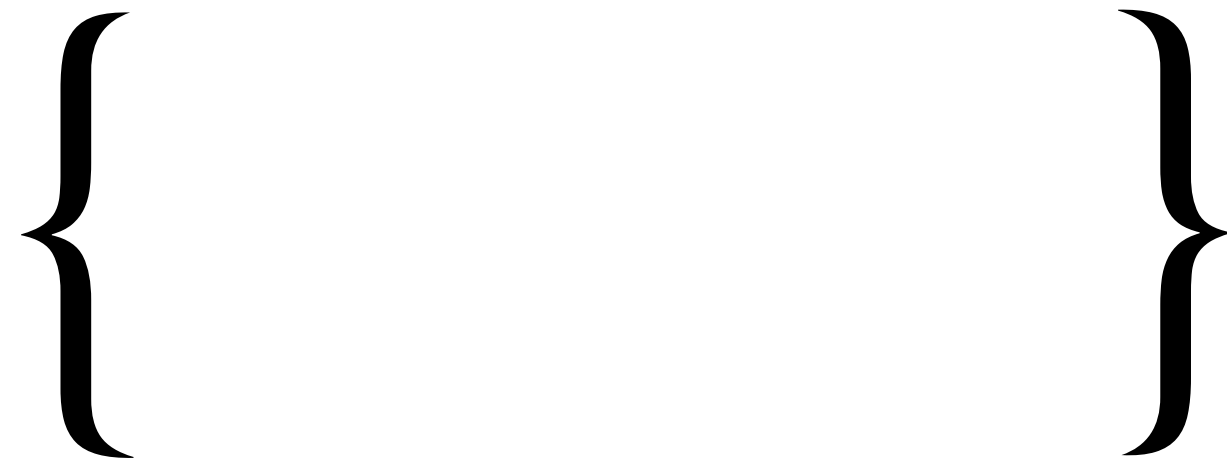


CSS

cool site stuff

What is CSS

- Cascading Style Sheet
- Made up of selectors and properties
- Creates rules that specify how the content of your HTML page should appear
- Web Browser matches selectors to HTML tags and applies properties
- CSS3



CSS properties always live within curly braces!

html { property: value }

CSS elements always have a selector being defined (html), an opening curly brace, the property being applied followed by a colon and the property's value, and a closing curly brace.

```
html {  
  property: value;  
  property: value;  
  property: value }
```

CSS elements sometimes have multiple **Properties** defined. These properties are separated by a **semicolon** after the property's value followed by the next property.

```
html, body, p {  
property: value }
```

CSS elements sometimes have multiple **Selectors** defined.
These selectors are separated by a single comma after the selector followed by the next selector. This applies the defined rules to each selector listed!

```
p {  
  color: blue  
}
```

selector property value

What do you think this piece of CSS does?

What CSS Selectors are available?

Universal Selector

* { } Targets EVERY element on the page.

Type Selectors

h1, h2, h3, html, body { } matches an elements
NAME

Class Selectors

.message { } matches an element that has a
defined class attribute (class="message")

p.message { } matches only p elements that
have a defined class attribute

ID Selectors

#heading { } matches an element that has a
defined ID attribute (id="heading")

div#message { } matches only div elements
that have a defined ID attribute

Descendant Selector

p a { } Targets EVERY <a> element that is
nested at any level under a <p> element

Child Selector

p > a { } Targets only <a> elements that are
direct children of any <p> elements

Next Sibling Selector

h1+p { } Targets only the first <p> element that
follows any <h1> element

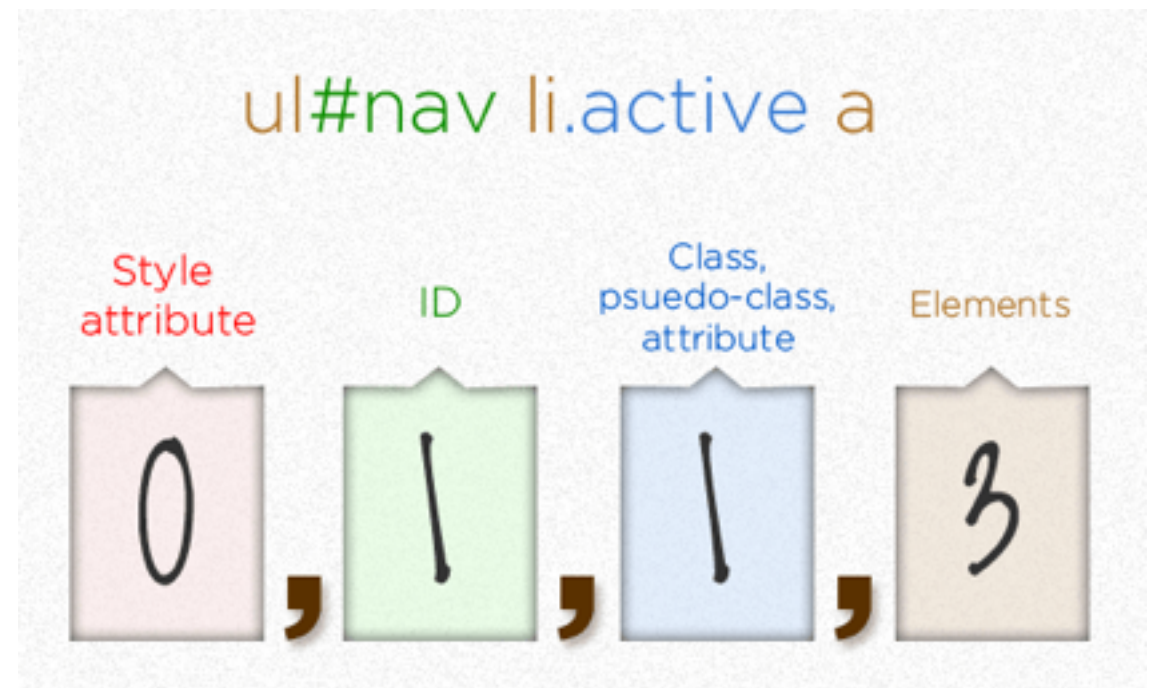
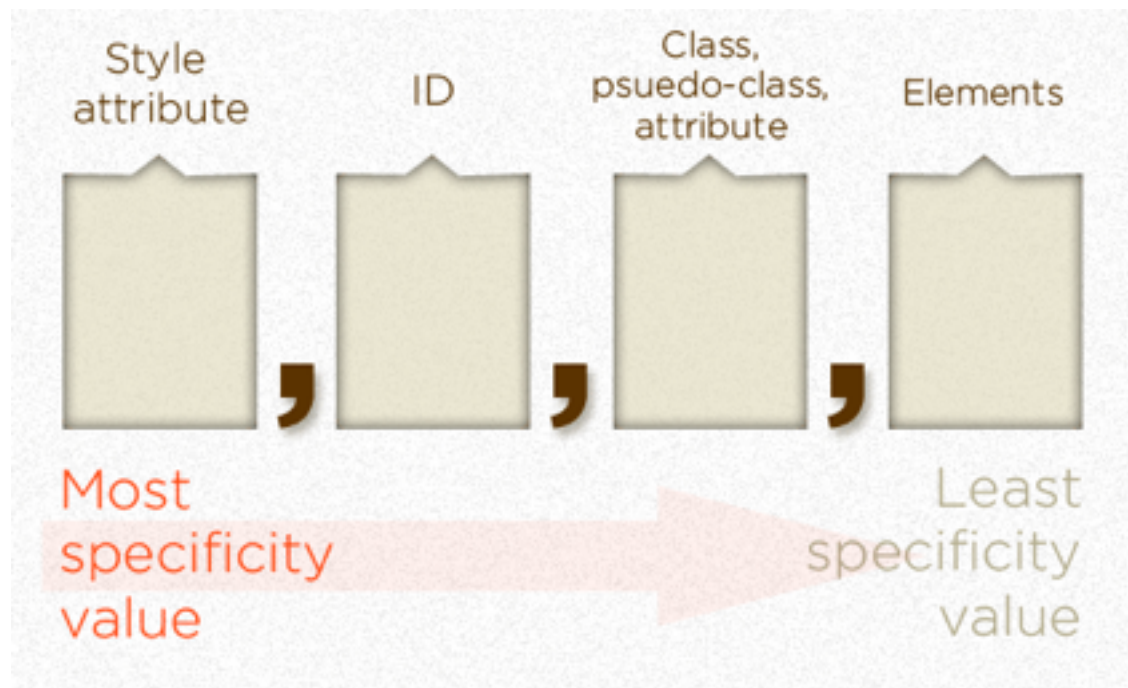
General Sibling Selector

h1~p { } Targets every <p> element that is a
sibling of any <h1> element

“Cascading”

- If two selectors are identical, the selector that comes **Last** takes precedent.
- `p { color: blue }`
`p { color: green }` What color would the text be?
- If a selector is more **Specific** than another on the page it will take precedent.
- `p.important { color: red }`
What color would a `<p>` element be now?
What color would `<p class="important">` be?

Specificity Calculated



Inheritance

- Some values of parent elements are **Inherited** by their children
- `body { font-family: helvetica }`
will apply the font family value to all of the body's children unless a more specific value is defined.
- **Not every property is inherited!**
Most values that are inherited by default pertain to text, fonts, and lists.

Color Properties

color: (green) (#00ff00) (0, 255, 0, 1)

The color property defines the Text color of an element. It can be a word, a hex value, or an RGBA value.

Text color

background-color: (red) etc.

The background color property defines the color of the element's "box"

Text color

opacity: 0.5

Opacity can be defined as it's own property or within the color property as (0, 0, 0, 0.5)

Text color

background-image: url('image.png')

**background-repeat: (repeat) (repeat-x)
(repeat-y) (none)**

background-size: (100% 100%) (cover) (x) (y)

Text Properties

font-family: 'Baskerville Old Face',

Garamond, serif

font-family defines the family to be used. Can be a specific font name in single quotes, a generic font name without quotes, or a style

Baskerville Garamond Georgia

font-weight: (bold) (light) (medium)

font-weight defines the weight of the font

Baskerville

font-style: (normal) (italic)

defines the font style

Baskerville

font-stretch: (regular) (extended)

(condensed)

Defines the version of the font to use.

Cond
ense Regular
d

font-size: (percent) (pixel) (em)

Defines font size in relative terms 100% 2em or absolute size 16px

Relative font sizes are inherited

Text Properties cont.

text-decoration: (none) (underline) (overline) (line-through)

Adds decoration to text

Baskerville

text-transform: (uppercase) (lowercase) (capitalize)

font-weight defines the weight of the font

BASKERVILLE

line-height: 1.5em

line-height = height of font + leading = 1em font + 0.5em leading

Baskerville

Baskerville

letter-spacing: 1em

Defines the tracking of the text (not kerning)

B a s k e r v i l l e

word-spacing: 1em

Defines the space between words.

Baskerville Baskerville Baskerville
Baskerville Baskerville Baskerville

Text Properties cont.

text-align: (left) (center) (right) (justify)

Text alignment

Baskerville font
family

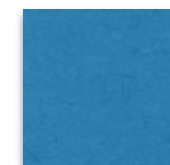
Baskerville font
family

vertical-align: (top) (middle) (bottom)

Aligns inline elements such as text to images



Baskerville font family



Baskerville font family

text-shadow: (-left/right) (-top/bottom) (blur) (color)

Text drop shadow. text-shadow: 0px 5px 0px green; would give a shadow centered and 5px down of the text, blurred 0px, and 100% green

Baskerville font family

Styling Links

- “a” selectors can be given additional pseudo-selectors to define states.
- a:hover
a:active
a:focus
- These give additional rules to the selected state (a different color on hover or focus)

Box Properties

width: (px) (em) (%)

height: (px) (em) (%)

By default, box level elements take up 100% width and the necessary height to fit their elements. You can set your own width and height this way.

min-width: (px) (em) (%)

min-height: (px) (em) (%)

The MINIMUM width and height that a box can be resized to. This is used when the page can shrink/expand based on the window size.

max-width: (px) (em) (%)

max-height: (px) (em) (%)

The MAXIMUM width and height that a box can be resized to. This is used when the page can shrink/expand based on the window size.

overflow: (hidden) (scroll)

If a box has a set height that is smaller than the content that it contains, you can tell the box what to do with that extra content, either hide the content (cut it off) or allow the box to scroll within itself.

margin: (top) (right) (bottom) (left)

padding: (top) (right) (bottom) (left)

margin-top:

margin-right: etc.

Defines the size of the margin or padding on the specified side.

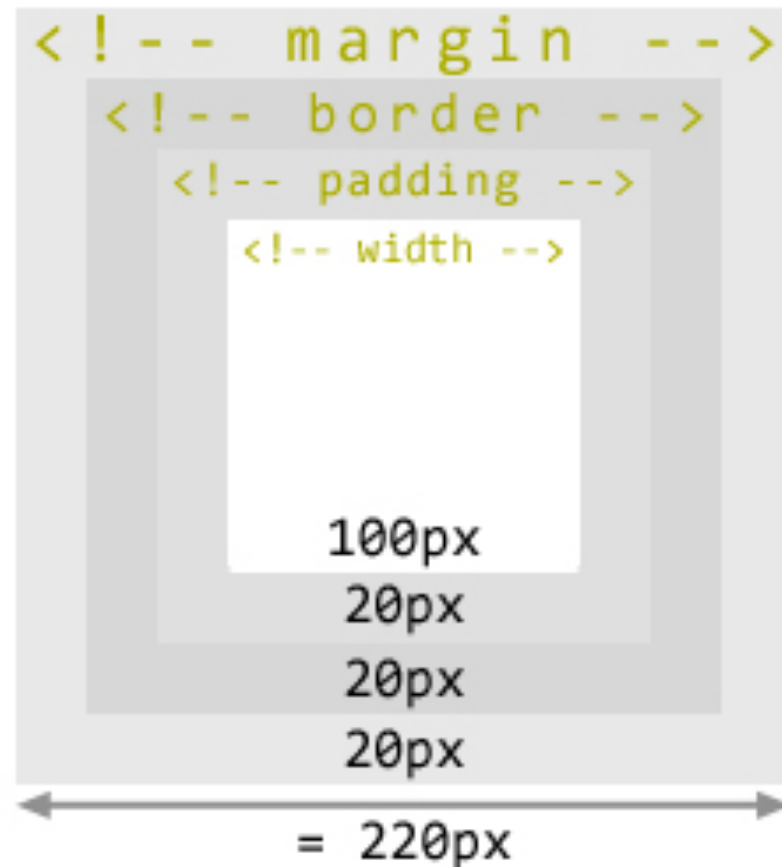
margin: 0 auto

Centers the content within it's parent by giving the left and right margin the value of auto.

Box Model: Padding/Margin

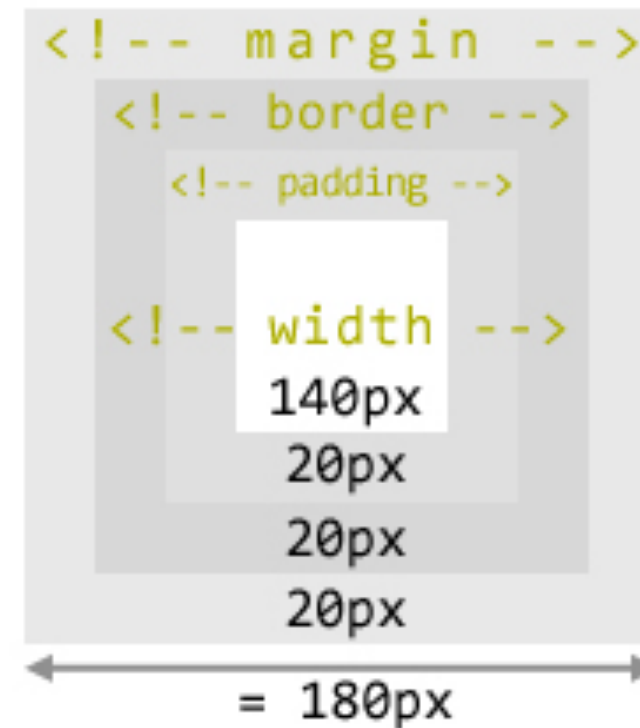
CSS Box Model

`box-sizing: content-box;`



`box-sizing: border-box;`

As opposed to the content-box model, the border-box model includes the border and padding inside of the width.



Box Properties

border: (width) (style) (color)

border: 2px solid black would give you a 2px solid black border around your content

Styles include: solid, dotted, dashed, double, groove, ridge, inset, outset, none

border-radius: (radius)

border-top-left-radius: (radius)

Give each corner a radius of __px or give specific corners a radius of __px

box-shadow: inset (-left/right) (-top/bottom) (blur) (spread) (color)

Acts like a text shadow but for a box. The word inset can also be added to inset the shadow instead of outset

border: (width) (style) (color)

border: 2px solid black would give you a 2px solid black border around your content



List Properties

list-style-type: (none) (disc) (circle) (square) (decimal) (decimal-leading-zero) (lower-alpha) (upper-alpha) (lower-roman) (upper-roman)

Style of list bullet point

- | | | | |
|--------|---------|---------|------|
| • List | 1. List | A. List | List |
| • List | 2. List | B. List | List |
| • List | 3. List | C. List | List |

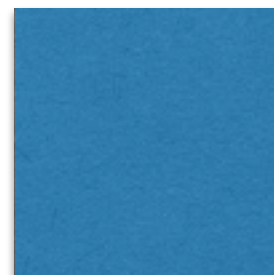
list-style-image: url('image.png')

- | | |
|-----------------|-----------------|
| • List
Style | • List
Style |
|-----------------|-----------------|

list-style-position: (outside) (inside)

box-shadow: inset (-left/right) (-top/bottom) (blur) (spread) (color)

Acts like a text shadow but for a box. The word inset can also be added to inset the shadow instead of outset



Layout Properties

display: (inline) (block) (inline-block) (none)

Display an inline element as a block or block level element as inline. inline-block is an odd case where the content is displayed as block but the container is displayed inline

position: (static) (relative) (absolute) (fixed)

Positioning of an element on the screen.

top:

bottom:

left:

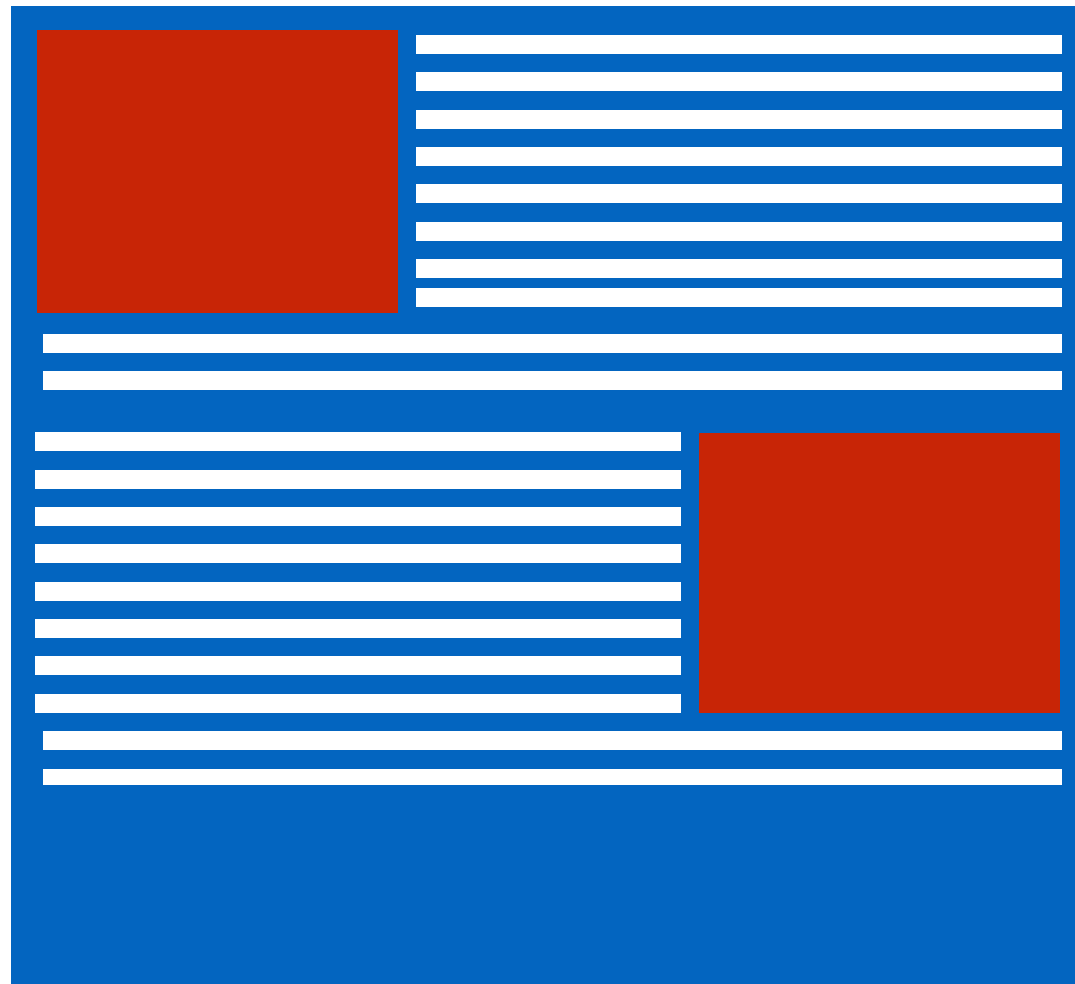
right:



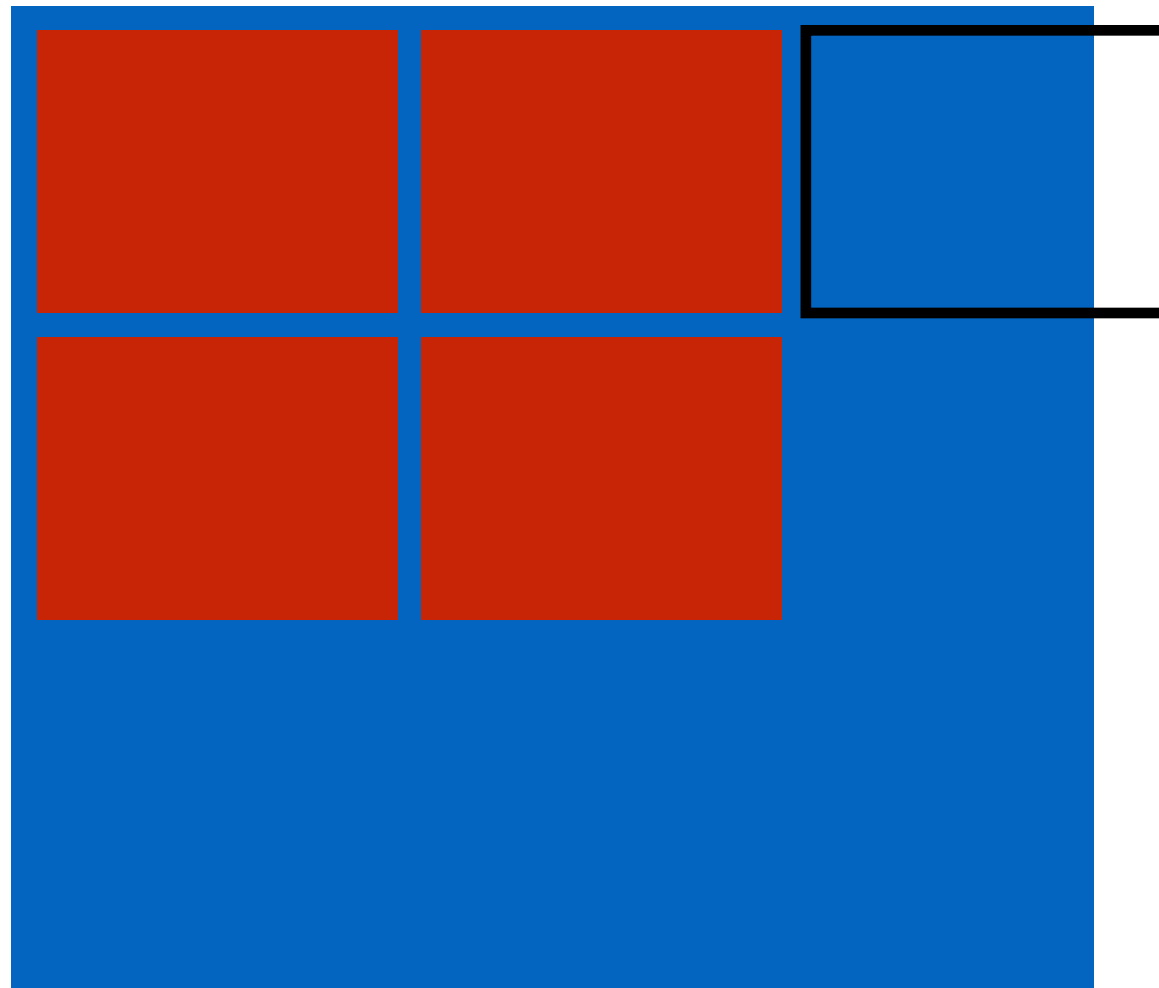
Float Properties

float: (left) (right) (none)

“Floats” an element to as far left or right as the containing element or other floating elements will allow. Floated elements need a defined width. Content will flow around floated elements



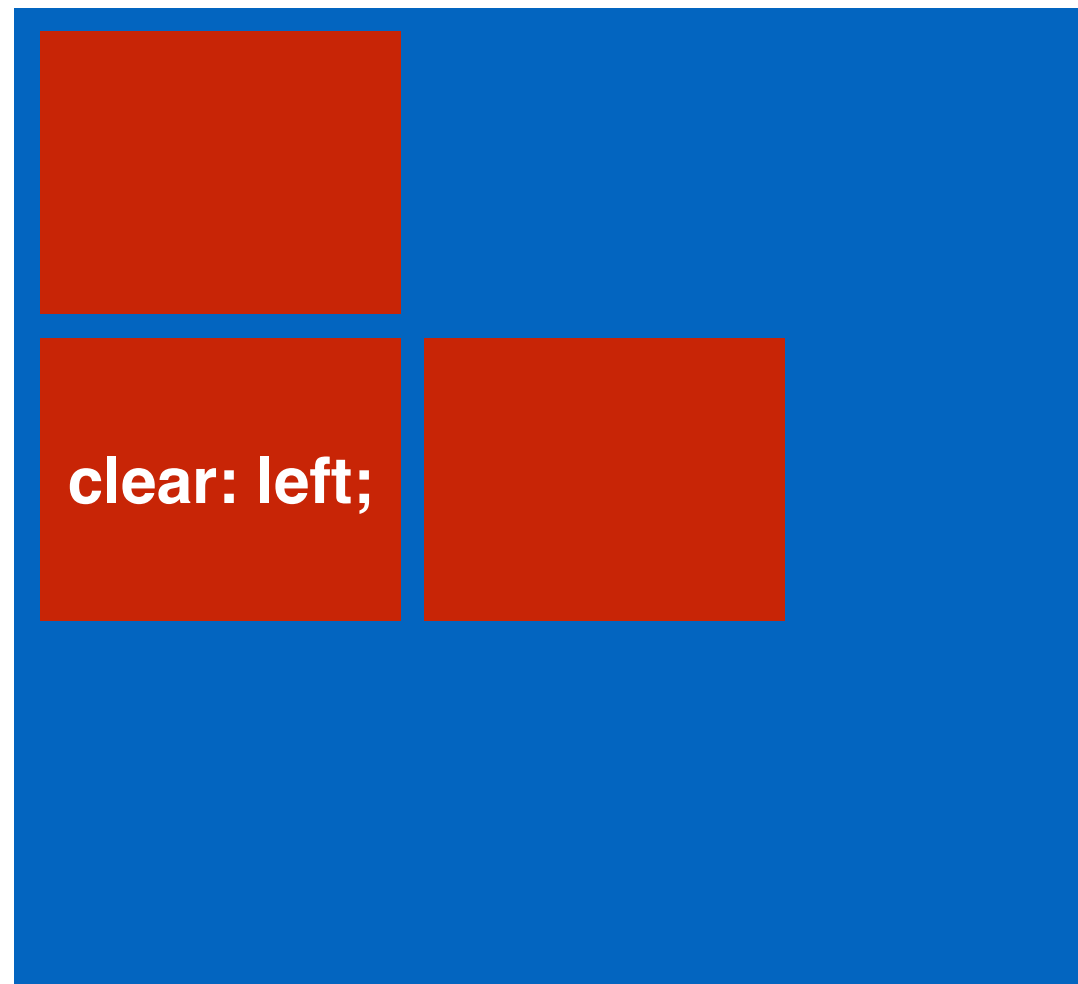
Float Properties cont.



Clearing Floats

clear: (left) (right) (both)

Floated elements will automatically float up to another floated element. In order to make floated elements drop to a new line you must clear the float on that side.



Parents of Floated Elements

- If a container contains ONLY floated elements, it will collapse onto itself and not see the element as having a height. To fix this, we apply a “clearfix” workaround



Clearfix

```
parent:before,  
parent: after {  
    display: table;  
    content: " ";  
}  
parent:after {  
    clear: both  
}
```

Fixed Width vs Liquid

- Fixed width layouts do NOT respond to page size. If you resize the window, a scrollbar will appear to show you the entire page
- Liquid layouts resize the content to fit the browser window. Resizing the window will shrink or expand the content to fit