## **CSS INTRO**

- Cascading Style Sheets (CSS)
- Made up of selectors and rules that define the visual style of elements
- CSS provides the rules to aesthetically style your webpage
  - Change fonts and the way text looks, colors
  - Width, height, background colors and images
  - Positioning
  - Lines and space between elements
- Rules cascade
  - An algorithm defining how to combine properties
  - Ones defined later override or build upon earlier ones
  - More specific ones override earlier ones
  - https://developer.mozilla.org/en-US/docs/Web/CSS/Cascade

# ADDING CSS TO YOUR PAGE

### Three main methods (only teaching 1)

- External Style Sheet
  - Text document with a .css extension
  - CSS File is linked to the HTML document in the head section
  - rel="stylesheet" href="style.css" />
  - Preferred way, separates presentation in another file

# ANATOMY OF A CSS RULE

```
declaration
selector { property: value; }
```

```
declaration block
selector {
  property1: value1;
  property2: value2;
  property3: value3;
}
```

### **CSS CONCEPTS**

- Inheritance
  - Some properties inherit their settings from their parent element
  - Mostly styles that effect text
- Parents & Children
  - Elements nested inside other elements are said to be children of that element.
  - The element that a given element is nested inside is its parent.
- It is very important that you understand the way your HTML is structured and how each of your elements are nested within each other to be successful writing complex CSS rules.

## **CSS INHERITANCE**

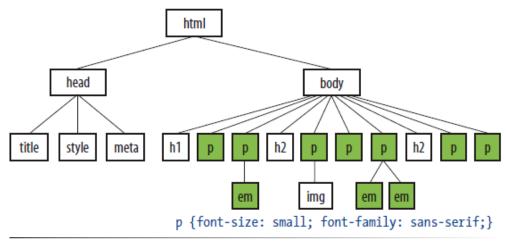


Figure 11-7. Certain properties applied to the p element are inherited by their children.

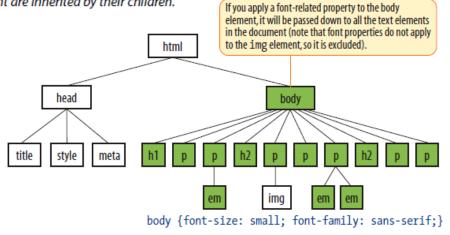


Figure 11-8. All the elements in the document inherit certain properties applied to the body element.

#### font-family

Values: one or more font or generic font family names, separated by commas | inherit

**Default:** depends on the browser

**Applies to:** all elements

Inherits: yes

Use the **font-family** property to specify a font or list of fonts (known as a font stack) by name, as shown in these examples.

```
body { font-family: Arial; }
var { font-family: Courier, monospace; }
p { font-family: "Duru Sans", Verdana, sans-serif; }
```

#### font-size

Values: length unit | percentage | xx-small | x-small | small | medium | large | x-large | xx-large |

smaller | larger | inherit

Default: medium

Applies to: all elements

Inherits: yes

font-weight

Values: normal | bold | bolder | lighter | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 |

inherit

Default: normal

Applies to: all elements

Inherits: yes

#### font-style

Values: normal | italic | oblique | inherit

Default: normal

Applies to: all elements

Inherits: yes

sample of oblique Times sample of true italic Times

This is an example of oblique Times as rendered in a browser.

This is an example of italic Times as rendered in a browser.

#### color

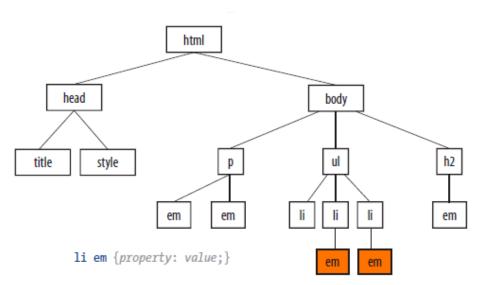


Figure 13-1. The 17 standard color names in CSS2.1.

alicellue 240,248,255 FOREIT	comstk. 255,248,220 FFF8DC.	darkturoudee 0,206,209 00CED1	notpank 255,105,180 PRESIDA	lightskyblue 135,206,250 83CEFA	midnightblue 25,25,112 191970	305,133,63 CD853F	snow 255,250,250 FFFAFA
antiquewhite 250,235,215 PAEBD7	220,20,60 DC14SC	dark-rolet 148,0,211 940003	Industred 205.92.92 CDSCSC	Hightslategray 119,138,153 778899	mintoream 245,255,250 PSFFFA	250.192.203 PPCDCB	opringgroun 0.255,127 00FF7F
0,255,255 00FFFF	Cyan 0,255,255 00FFFF	despunk 255,29,147 (#149)	indigo 75,0,130 490683	lightsteefblue 176,196,222 ROCADE	mistyrose 255,228,225 FFE4F1	pkm 221,160,221 00A000	50,130,180 46,82,84
aquamarine	darkblue	deepskyhise	lvory	lightyellow	maccisin	powderblue	184
127,255,212	0,6,139	0,191,255	255,240,240	255,255,224	250,226,181	176,224,230	210,100,140
7FFD4	000008	000777	FFFBF0	FFFED	FFE485	800066	DZD40C
240,255,255 PGFFFF	Gerkizyan 0,139,139 008889	dengrey 105, 105, 105 113, 61,61	kheki 240,230,340 PODSBC	0,755,0 0,755,0 00FF00	navejowhite 255,222,178 PFDEAD	purple 128,0,128 800000	356 0,126,126 000000
beige 245,245,220 PSFSDC	darkgolderros 184,134,11 800608	dodgerblue 30,144,250 LEGGFF	lawender 230,230,250 8685FA	imagraen 56,205,98 30CD32	0,0,125 000060	PEG 220,0,0 FF6000	trictle 210,191,210 DBBFDB
bleque 255,228,196 FFERC4	derkgray 169,169,169 AJAJAJ	Arebnick 178,34,34 822222	lavenderblush 255,240,245 PPPDF5	linen 250,240,230 FAFOE6	oldlece 253,345,230 PDF566	168,143,143 BESTEF	tometo 253.96,71 Pr6347
black	darkgreen	toreiwhite	lawingrees	magenta	0098	royaliskue	turquose
0,0,0	0,100,0	255,250,240	124,252,8	255,0,255	128,128,0	65,105,225	64,224,208
000000	006400	FFFAFO	2CPC00	procep	808000	4169E1	408000
blandhedalmond	darkishaki	forestyroan	lemondhaften	120,000	Offvedratio	Gaddlebrown	wolet
250,250,255	189.183.107	34,139,34	255,250,205	120,000	107,142,35	139,69,19	238,130,238
PPFFCD	808768	228672	PPEACD	100,000	058123	884513	818212
blue	darkmagenta	10/0164	IngRicolate	mediumaquamanno	orence	Samon	White
0,0,25\$	139,0,139	255,0,255	173,216,230	182,305,170	255,965,0	250,128,114	255,255,255
0000FF	880088	FF00FF	ADDRES	66CDAA	FFA500	FAE072	PPFFFF
blueriolet 136,43,226 6A26E2	darkollvigrees 85,107,47 55082F	gainsboro 220,220,220 DCDCDC	lightcoral 240,128,128 r08080	0,0,205 0000CD	onchid 218.112,214 DA7006	sandybrown 364,164,96 74A460	wheat 245,222,179 FSDE03
165,42,42 A55A9A	derkorange 255,140,0 978000	ghostwhite 248,248,255 ranser	lightgolderrodyclow 250,250,210 PATAD2	mediumorchid 186,85,211 8ASS01	orangened 255,69,0 FF4500	56837001 46,139,87 200057	whitesmoke 245,245,245 especies
hurlywood	darkred	gold	lightryan	mediumpurple	palogolderend	soathell	yellow
222,184,135	139,0,0	255,215,0	224,259,250	147,612,219	236,232,170	250,245,236	255,255,0
DED887	8B3000	170700	EOFFFF	937000	EEESAA	FFFSEE	FFF00
Cadettilus IIS, 158, 160 3190AU	derkorchid 153,50,204 9902CG	golderod 218,185,12 DAASIU	lightgreen 144,218,144 90ez90	modlumasagreen 60,179,113 3053/1	palogreen 152,351,152 987896	150,82,45 A05220	yellowgreen 154,205,50 9ACD34
chartmuse	darksalmon	gray	lightgrey	mediumsiateriue	paletarquoise	silver	
127,258,0	233,150,122	126,129,128	211,211,211	123,104,238	175,238,238	192,192,192	
7FFF00	85067A	808080	030103	766800	AFEEE	COCDCO	
checylote	derkenogreen	green	lightpirik	mediumspringgreen	palewioletrad	akyblus	
210,005.30	143,188,143	0.128.0	255,182,193	0,250,154	219,112,147	135,206,235	
02091£	BFBCBF	008000	PFB6C1	00FASA	DB7093	87CEEB	
coral	derkslateblue	great/felow	lightsalmon	mediumturquoise	Dapayowhia	stateblue	
255,127,60	72,61,139	173,255,47	255;160;122	72,209,204	255,239,213	106,90,205	
27750	483068	ADFF2F	95A07A	46D1CC	FFEFDS	6ASACD	
comflowerslass	derkeletegray	honoydew	lightsebgreen	mediumvioletred	peachpuff	claregray	
100,149,237	47,79,79	240,255,240	32,178,179	199,21,153	255,239,213	112.128,144	
6493ED	2F4F4F	FOFFF0	2082AA	CT1385	FFEFOS	708090	

```
Element selector p { color: navy; }

Grouped selectors p, ul, td, th { color: navy; }
```



It is also possible to nest descendant selectors several layers deep. This example targets em elements that appear in anchors (a) in ordered lists (o1).

```
ol a em { font-variant: small-caps; }
```

#### Child selector

A child selector is similar to a descendant selector, but it targets only the direct children of a given element. There may be no other hierarchical levels in between. They are indicated with the greater-than symbol (>). The following rule affects emphasized text, but only when it is directly contained in a p element. An em element inside a link (a) within the paragraph would not be affected.

```
p > em {font-weight: bold;}
```

#### Adjacent sibling selector

An adjacent sibling selector targets an element that comes directly after another element with the same parent. It is indicated with a plus (+) sign. This rule gives special treatment to paragraphs that follow an h1. Other paragraphs are unaffected.

```
h1 + p {font-style: italic;}
```

#### General sibling selectors

#### NEW IN CSS3

A general sibling selector selects an element that shares a parent with the specified element and occurs after it in the source order. They do not need to follow one another directly. This type of selector is new in CSS3 and is not supported by Internet Explorer 8 and earlier. The following rule selects any h2 that both shares a parent element (such as a section or article) with an h1 and appears after it in the document.

```
h1 ~ h2 {font-weight: normal;}
```

# **CSS SPECIFICITY & BOX MODEL**

### **CSS CASCADE & SPECIFICITY**

- Style passes down (cascades) until a rule with more weight overrides a previous style
- First goes by style sheet hierarchy
- Then goes by order defined in the style sheets
- If there is a conflict it resolves with a point system
  - Creators developed a point system
  - inline is 1000, id is 100, class is 10, element is 1
- See Charts on next two slides
- https://developer.mozilla.org/en-US/docs/Web/CSS/Cascade
- https://developer.mozilla.org/en-US/docs/Web/CSS/Specificity
- http://css-tricks.com/specifics-on-css-specificity/
- http://www.smashingmagazine.com/2007/07/27/css-specificity-things-you-should-know/
- http://code.tutsplus.com/tutorials/quick-tip-understanding-css-specificity--net-10963

# **CSS SPECIFICITY**

CSS Selector Specificity - Cheat Sheet

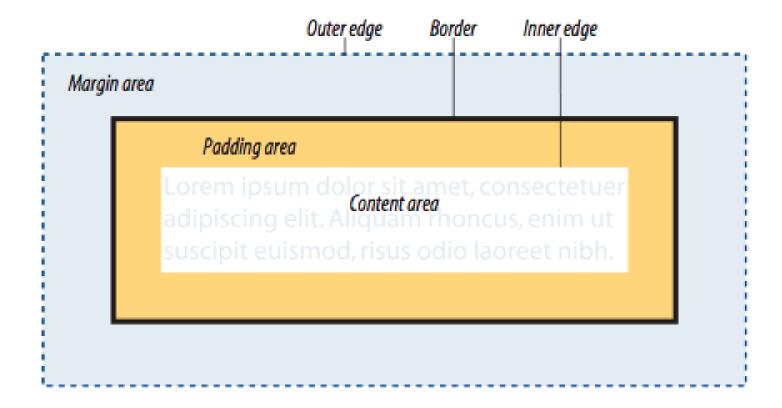
	*	id	classes	elements	result*
	a	b	c	d	
li	0	0	0	1	1
.lorem	0	0	10	0	10
#ipsum	0	100	0	0	100
style = " "	1000	0	0	0	1000
ul li	0	0	0	2	2
ul ol + li	0	0	0	3	3
ul ol li.red	0	0	10	3	13
li:first-line	0	0	0	2	2
#ipsum li	0	100	0	1	101
a:link	0	0	10	1	11

#### Legend:

a=1000	style = " "	*Bigger
b=100	ID attributes	is
c=10	Classes and pseudo-classes	more
d=1	Elements and pseudo-elements	specific

### **CSS BOX MODEL**

The Browser sees every element, block or inline, as a little rectangular box



### **CSS BOX MODEL**

#### Standard CSS Box Model

- This is the standard way the box model is calculated.
- CSS box-sizing property is:
  - Not set
  - Set to content-box
- Width of an element on screen is:
  - width + padding + border
  - width property sets content area width
  - You must account for padding and borders when setting the width to get the exact size you want.
- Supported in CSS 1
- https://developer.mozilla.org/en-US/docs/Web/CSS/box\_model
- http://css-tricks.com/the-css-box-model/

### **CSS BOX MODEL**

#### **New Border Box CSS Box Model**

- This is the new way the box model can be calculated.
- CSS box-sizing property is:
  - Set to border-box
- Width of an element on screen is:
  - Whatever you set the width property to.
  - The browser will take the width of the padding and border out of the content area so your final box is the size you set.
- CSS 3 Property that has fairly good browser support
  - http://caniuse.com/#feat=css3-boxsizing
  - Even back to IE8
- http://www.paulirish.com/2012/box-sizing-border-box-ftw/
- https://developer.mozilla.org/en-US/docs/Web/CSS/box-sizing
- http://css-tricks.com/box-sizing/

# **NAVIGATION**

# CSS HORIZONTAL NAVIGATION

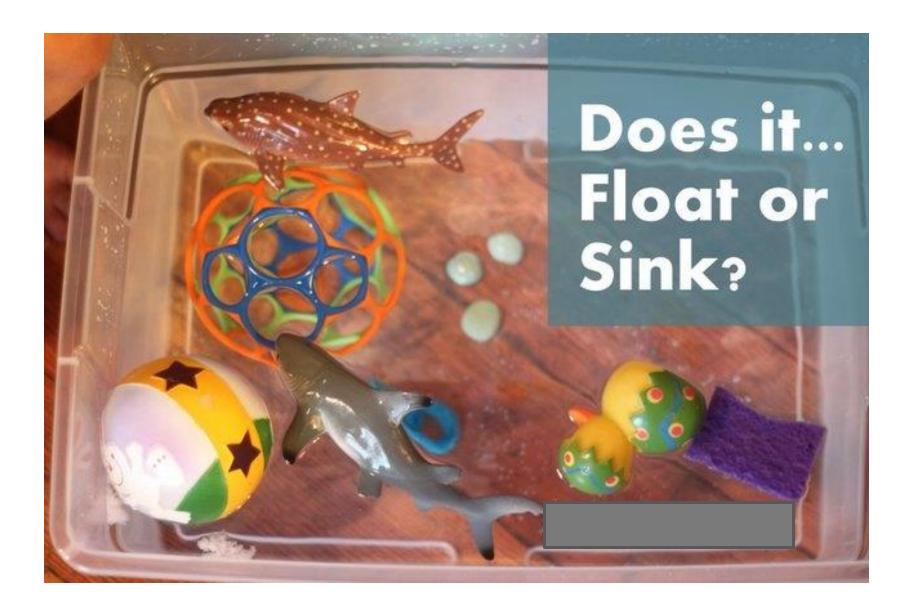
- Navigation is a list of links so we should use a list as the html element
- ul
  - Remove the bullets list-style-type: none
  - Remove the margins and padding
- ul li
  - Float list items left float: left;
- ullia
  - Make anchor block and apply styles display:block;
- Clear the element that follows clear:left;

# **CSS RESETS**

### **CSS RESETS**

- Used to reset all css properties so they will be consistent in all browsers.
- Eric Meyer's css reset
  - Complete reset to no styles
  - http://meyerweb.com/eric/tools/css/reset/
- Normalize
  - Targets only styles that need to be reset
  - http://necolas.github.com/normalize.css/
- Yahoo YUI css reset
  - Mostly complete reset
  - http://yuilibrary.com/yui/docs/cssreset/
- http://www.cssreset.com/

# **CSS FLOATS**



### **CSS FLOATS**

- "The float CSS property specifies that an element should be taken from the normal flow and placed along the left or right side of its container, where text and inline elements will wrap around it. A floating element is one where the computed value of float is not none." – Mozilla MDN
- https://developer.mozilla.org/en-US/docs/Web/CSS/float
- Float value can be left, right, or none. Value of none turns off float
- Float property causes element to move as far as possible to the left or right and allow other elements to wrap around it.
- Always provide a width to floated element
- Floated elements behave like block elements
- Margins do not collapse like they do on a standard block element (top and bottom margins)

### **CSS FLOATS**

- Clearing the float.
  - Apply the clear property to the element following the float that you don't want to wrap around the content.
  - It tells the element it is on that it can not be positioned next to a floated element.
- If a containing element only contains floated elements it will have no height unless you set some other properties.
  - Search clearfix
  - Or use overflow property set to hidden.
- You can use floats to create column layouts
- Floated elements need to appear in html before content that wraps around them

# **CSS POSITION**

## **CSS POSITION**

#### 4 values for CSS positioning (position property)

- Static
  - Normal default position in the document flow
- Relative
  - element is moved relative to position in document flow
  - Space in document flow is reserved
- Absolute
  - Removed from the document flow
  - Positioned with respect to containing element with a position
- Fixed
  - Removed from flow and positioned fixed to the window
- Then use top, bottom, left, or right properties to set position
- z-index property sets render layer. Only works when element has position set