

HTML & CSS Level 1: Week 3

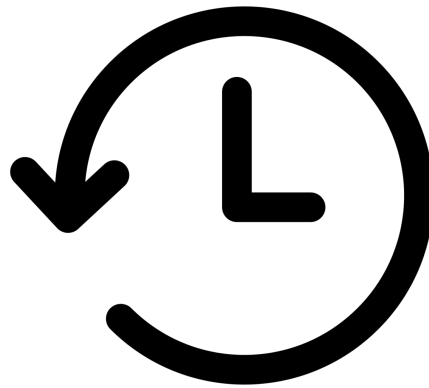
November 18, 2014

Instructor: Devon Persing

This week

- Web fonts
- More kinds of CSS selectors
- More background styles
- HTML5 containers

Review!



History designed by [Ema Dimitrova](#) from the [Noun Project](#)

<> Block and inline elements

- Block elements (ex.: **<div>**)
 - Take up space
 - Have height and width
 - Create line breaks
- Inline elements (ex.: ****)
 - Flow with content
 - Take up only as much space as they need
 - Don't create line breaks
- Inline-block elements (ex.: ****)
 - Two great tastes in one

{ } Anatomy of a CSS rule

selector { property: value; }

- **Selector** is the thing you want to style
- **Property** is what aspect you want to style
- **Value** is how you want to style it
- **Property + value = declaration**
- **Declarations** end in semicolons (;)

{ } CSS rule example

```
h1 { font-size: 2em; }
```

- **Selector** is `h1` (any `<h1>` on the page)
- **Property** is `font-size`
- **Value** is `2em`

{ } Type selectors

- **Type selectors** are single HTML element names that style all elements of that type on the page

```
h1 { font-size: 2em; }
```

{ } Descendent selectors

- **Descendent selectors** point to *children* of other selectors and are *more specific*

```
/* less specific type selector will style all <a>  
elements */
```

```
a { font-weight: bold; }
```

```
/* more specific type selector will style only <a>  
elements that are children of <p> elements */
```

```
p a { font-weight: normal; }
```


{ } Multiple selectors

- Rules can have **multiple selectors**
- Selectors can be of **mixed kinds**

```
/* single selector */
```

```
a { font-weight: bold; }
```

```
/* multiple selectors */
```

```
a, span { font-weight: bold; }
```

```
/* multiple selectors with a descendent selector */
```

```
p a, span { font-weight: bold; }
```

{ } Multiple declarations

- Rules can have **multiple declarations**

```
/* single declaration */
```

```
a { font-weight: bold; }
```

```
/* multiple declarations (and multiple selectors!) */
```

```
a, span {  
    font-weight: bold;  
    font-style: italic;  
}
```

{ } Common font & text properties

font-size

text-align

font-family

text-transform

font-style

text-decoration

font-weight

Lots of others!

line-height

letter-spacing

Lots of others!

{ } Colors

`color`

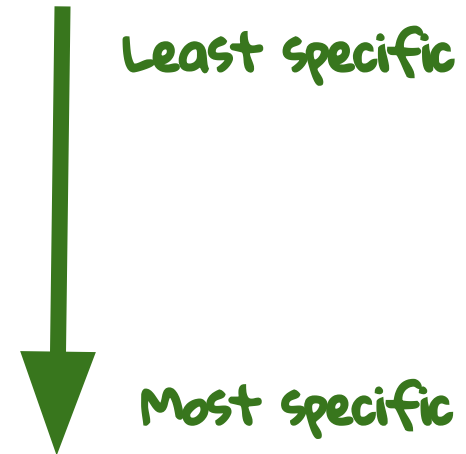
`background-color`

Values for `color` and
`background-color`:

- Hex: `#ff0000`
- RGB: `rgb(255,0,0)`
- *And others!*

{ } Stylesheet "location"

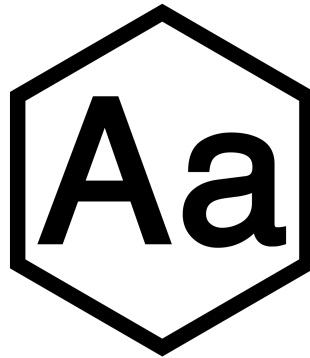
- Styles that are "closer" to the elements they style take precedence
 - Browser defaults
 - External styles (in a `.css` file)
 - Internal styles (in the `<head>`)
 - Inline styles (in an element)



{ } Overriding styles

- Declarations that target children are more specific than rules children inherit from parents
- Declarations that come later override rules that come earlier
- **Hint:** Web Inspector will list the most specific styles on top and cross out overridden styles

Web fonts



{ } Freedom from Arial!

- Web fonts let us style sites with **fonts that users may not have** on their own device
- Web font services **licence fonts for online use** specifically
- Files are either:
 - hosted by a service
 - served with your pages



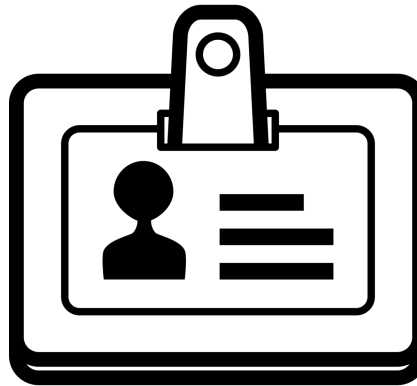
A note about licensing

- **Not all fonts can be used online**, even if you own their rights for print, they're in Adobe products, etc.
- Fonts with online licensing will come with **documentation saying so**
- **Exception:** If you own the rights to use a font with software, you can use it to make images that are published online

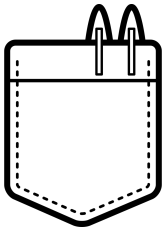
{ } Some web font options

- **Google Fonts** is free and hosted
- **TypeKit** (owned by Adobe) is hosted and subscription based or bundled with Creative Cloud
- **FontSquirrel** is free and not hosted
- **FontDeck** is subscription based and not hosted
- *And some others!*

Using class and id selectors



Identification Badge designed by [Michela Tannoia](#) from the [Noun Project](#)



Combining concepts

- **Week 1:** HTML elements can have attributes
- **Week 2:** HTML element names can be used as CSS selectors
- **Week 3:** HTML attributes can *also* be used as CSS selectors

</> / {} **class and id**

- **class** and **id** attributes can be added to any HTML element
- **Classes** are for multiple things on a page
- **IDs** are for individual things on a page
- You can make up whatever **class** and **id** values you want!

<> class attributes in HTML

- Classes can be shared by multiple elements on a page

```
<h1 class="kittens">...</h1>
```

```
<span class="kittens">...</span>
```

- Elements can have multiple classes

```
<div class="kittens puppies">...</div>
```

```
<div class="kittens puppies birds">...</div>
```

{ } class selectors in CSS

- Start with a **period** (.)
- Can style any element with the class

```
.kittens { color: #000000; }
```

- Or can be used to style only a specific **type** of element with the class

```
h3.kittens { color: #000000; }
```

- More specific than an HTML type selector

<> id attributes

- IDs *cannot* be shared by multiple elements on a single page
- Elements *cannot* have multiple IDs

```
<div id="kittens">...</div>
```

```
<div id="puppies">...</div>
```

```
<div id="birds">...</div>
```


{ } id selectors in CSS

- Start with a **hash/pound sign (#)**
- Can style the single element with the ID

```
#kittens { color: #000000; }
```

- More specific than a class selector

Mixing class and id attributes

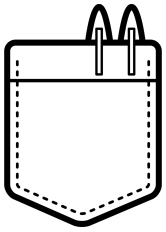
- Elements can have **id** and **class** attributes at the same time

```
<div id="kittens">...</div>
```

```
<div id="puppies" class="small floppy">...</div>
```

```
<div id="birds" class="small feathery">...</div>
```

- ID selector styles can be used to override class selector styles



Other uses for `ids`

- Pre-HTML5, used to "label" major areas of the page

```
<div id="header">...</div>
```

```
<div id="article">...</div>
```

```
<div id="sidebar">...</div>
```

```
<div id="footer">...</div>
```

- Used to make on-page links

```
<a href="#header">Go back to the top</a>
```



Be thoughtful in your selectors

- Recommended order of attack:
 - a. Type selectors
 - b. Class selectors
 - c. Descendent selectors
 - d. ID selectors
- If you overuse IDs in your styles, **you're going to have a bad time**

CSS pseudo-classes and pseudo-elements

{ }

{ } Pseudo-classes are conditional

- **Pseudo-classes** are added to a selector to add conditional styles to an element
- Most commonly used to style **states** of **<a>** elements and form elements

```
a:link { /* the default state of a link */ }
```

```
a:visited { /* a link that's been clicked */ }
```

```
a:hover { /* a link that has a mouse hover */ }
```

```
a:focus { /* a link that has keyboard focus */ }
```

```
a:active { /* a link that is being clicked */ }
```

{ } :hover versus :focus

- **:hover** is for a link or other interactive element that has a **mouse hover**
- **:focus** is for a link or other interactive element that has **keyboard focus**
- Browsers have their own default **:focus** styles for **accessibility**

```
a:hover, a:focus {
```

```
/* it's good practice to style them together! */
```

```
}
```

{ } :hover for other elements

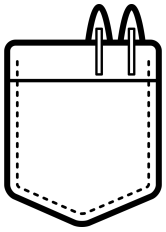
- **:hover** can be used to style hover states for some non-interactive elements to create a more dynamic experience

```
tr { /* a table row with one background... */  
    background: #9f6;  
}
```

```
tr:hover { /* ...could have another on hover */  
    background: #f60;  
}
```


{ } Some nifty pseudo-things

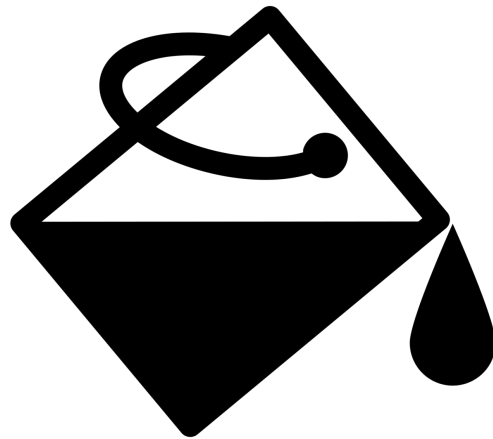
- **:first-letter** styles the first letter of a block of text
- **:first-child** and **:last-child** style the first and last children of a parent
- **:nth-child()** can be used to style even or odd children, or do some math to style every 5th, etc.
- **::selection** styles text that is selected by the user



CSS selectors are evolving

- **Pseudo-classes, pseudo-elements, combinators, and attribute selectors** create extremely targeted ways to style content that degrade gracefully in older browsers
- To learn more of these techniques: <http://www.quirksmode.org/css/selectors/>

More background styles



Paint Can designed by [Alex Valdivia](#) from the [Noun Project](#)

{ } background-color review

```
.block {  
    /* text is black and centered */  
    color: #000000;  
    text-align: center;  
    /* background is light mauve (#bc7384) */  
    background-color: rgb(188,115,132);  
}
```



I'm covering up a kitten.
:(

{ } Transparent background-color

```
.block {  
    /* text is black and centered */  
    color: #000000;  
    text-align: center;  
    /* background is light mauve and 50% opaque */  
    background-color: rgba(188,115,132,0.5) ;  
}
```



I'm partially covering up
a kitten. :|

{ } Styling a background image

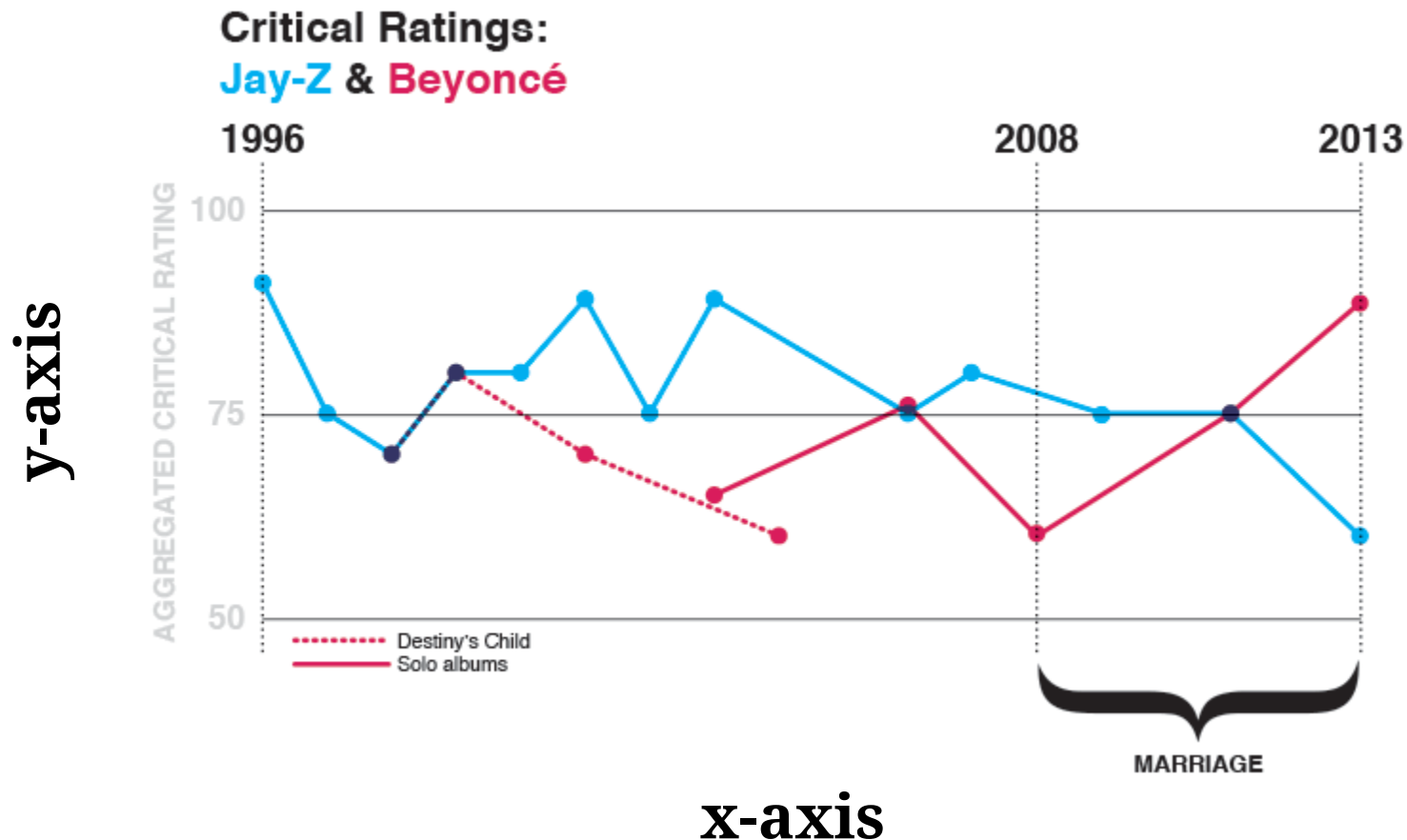
- The property is **background-image**
- The value is a **URL where an image lives**

```
.kittens {  
    background-image: url("img/kittens.jpg");  
}
```

{ } Styling a background image

- **background-repeat:** repeat horizontally or vertically, or not at all
- **background-position:** Start at the left or right, top or bottom, and center or not
- **background-attachment:** Does it stick or scroll
- **background-size:** How much of the container does it cover

{ } Left & right, top & bottom



{ } Repeating a background

```
/* repeat the background horizontally */
```

```
background-repeat: repeat-x;
```

```
/* repeat the background vertically */
```

```
background-repeat: repeat-y;
```

```
/* don't repeat the background */
```

```
background-repeat: no-repeat;
```

{ } Positioning a background

- **background-position** values include both the x-axis and y-axis
- x-axis first, y-axis second
- Can be **left/right top/bottom** *or* any measurement (pixels, %, ems, etc.)

```
/* position a background in the left top corner */
```

```
background-position: left top;
```

{ } Positioning a background ex.

```
/* position a background in the left top corner */
```

```
background-position: left top;
```

```
/* positioning a background in the right bottom corner */
```

```
background-position: right bottom;
```

```
/* position a background on the left, centered vertically  
*/
```

```
background-position: left center;
```

```
/* position an image completely centered */
```

```
background-position: center;
```

{ } Background attachment

`/* have the background scroll (the default) */`

background-attachment: scroll;

`/* have the background stick regardless of scrolling */`

background-attachment: fixed;

...and some others

{ } The magical image background

```
/* make a full-sized, fixed image background that covers  
the whole container */
```

```
.magic-bg {  
    background-image: url(img.png);  
    background-repeat: no-repeat;  
    background-position: center center;  
    background-attachment: fixed;  
    background-size: cover;  
}
```

"Homework" for next time

1. Practice with image backgrounds
2. Refine your design with a wider variety of CSS selectors
3. Have a great holiday!

Optional: HTML5 for Web Designers, ch. 1 & 5

Optional: HTML5 & CSS, ch. 16, reread 10 for selectors

Next time:

- Review!
- CSS abbreviations
- CSS box model
- Positioning and layouts with CSS
- Resetting browser default styles
- **New location:** 2300 7th Ave, Suite B



Have a
great
break
next
week.

[Turkey on Flickr](#)

Questions? Comments?

- Visit dpersing.github.io/svc for:
 - Class slides
 - Code samples
 - Resources
- Email me: dep@dpersing.com
- Tweet at me: [@devonpersing](https://twitter.com/devonpersing)