HTML, CSS, & Fundamentals of Development

School of Visual Concepts
Scott Thiessen, instructor

http://movingobjects.io/svc/
scott@movingobjects.io

Week 5

- Fun CSS Tricks
- Embedding Audio & Video
- Preview: Responsive Design
- Preview: JavaScript, CMS's, Frameworks
- Paths Ahead: Apps, Games, and Artwork
- Class Projects

Recap

HTML Tags

```
<h1>      <strong>            <ul>
<h2>      <em>                <ol>
<h3>                          <li>
          <a href="">
<p>       <img src="">
```

HTML Tags

<main>

<header>
<footer>

<section>
<article>

<div>

CSS Review

Typography

color
font-family
font-size
line-height
letter-spacing

font-style
font-weight

text-align
text-transform
text-decoration

Sizing

width
height

Spacing

margin
padding

Backgrounds

background-color
background-image
background-repeat
background-size

Borders

border
border-radius

Special Selectors

:hover
:active

CSS Review

```css
h1 {
    font-size: 32px;
}

body {
    background-color: black;
    font-family: serif;
    color: white;
}
```

ID & Class

```
<p id="intro">
   The only intro paragraph.
</p>

<p class=“highlighted”>
    One of many possible
    highlighted paragraphs.
</p>
```

Select HTML Element

Use the element (tag) name

`<body>`	`body { }`
`<h1>`	`h1 { }`

Select by ID

Use the # sign

```
<div id="top">          div#top { }
                        #top { }

<ul id="nav">           ul#nav { }
                        #nav { }
```

Select by Class

Use a . (dot)

```
<a class="selected">    a.selected { }
                        .selected { }

<h1 class="special">    h1.special { }
                        .special { }
```

Select "descendent"

Use space between two selectors

```
<ul>                       ul li { }
  <li>Item</li>
</ul>

<h1>                       h1 em { }
  <em>Hi</em> there!
</h1>
```

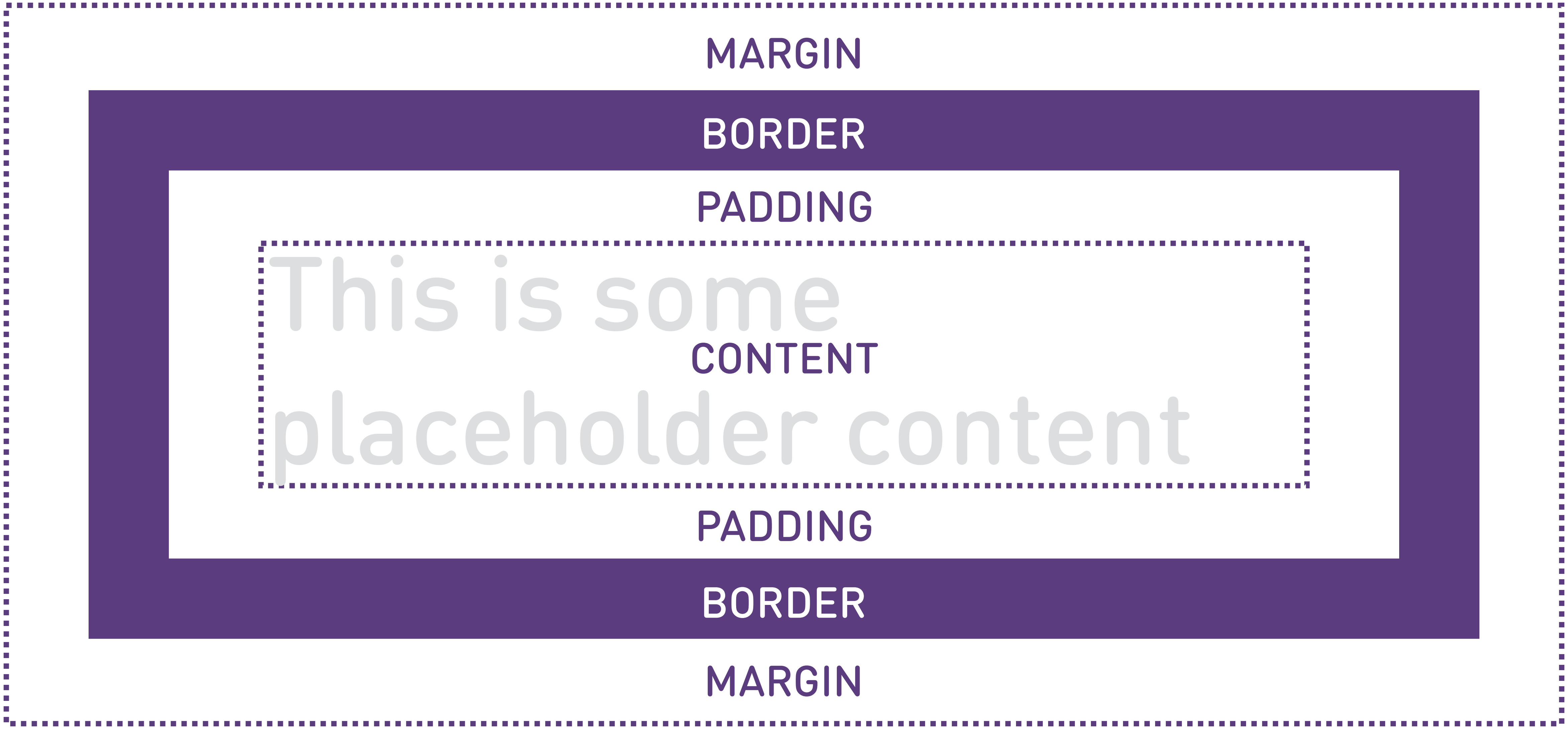
CSS Selectors

Mix & Match

```
ul#nav li a { }
```

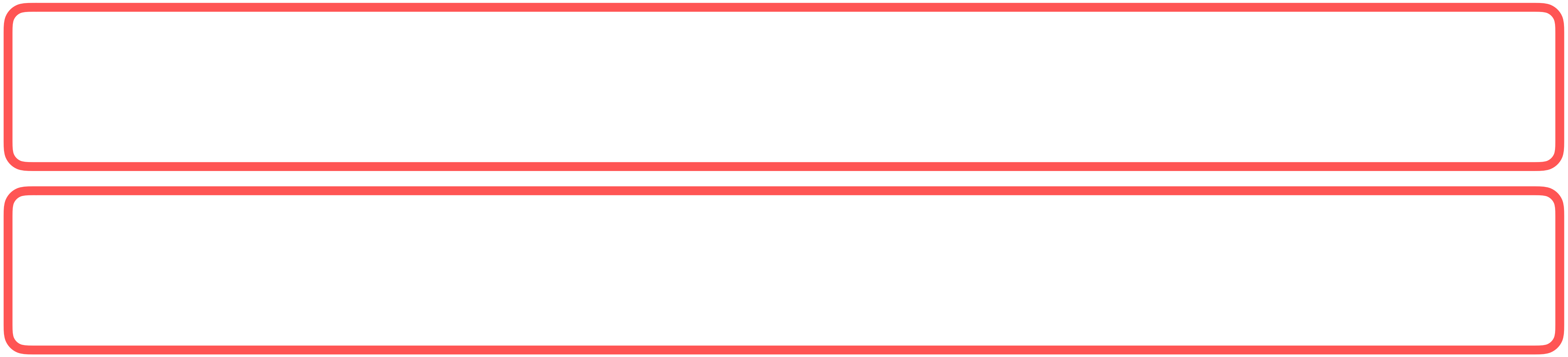
```
<ul id="nav">
  <li>
    <a href="#">Link</a>
  </li>
</ul>
```

The Box Model

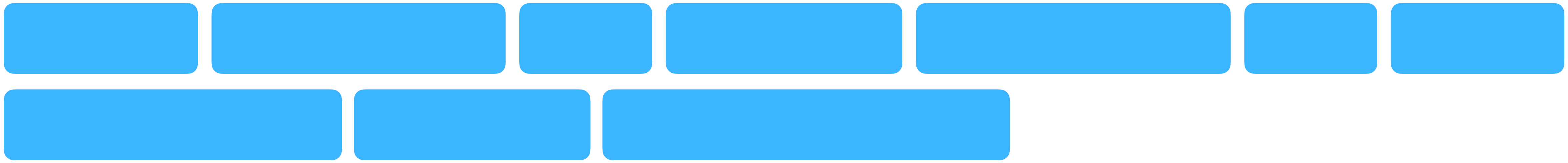


Block vs Inline elements

Block



Inline



Positioning methods

- `relative`: position in relation to the flow of the page
- `absolute`: position in a specific place outside of the flow of the page
- `fixed`: position relative to the browser window, unaffected by scrolling

How it works

You set (a) the positioning method, and (b) the distance from the left/right and top/bottom in CSS.

```
div {
   position: absolute;
   left: 25px;
   top: 25px;
}
```

Process Overview

- Put content in HTML
- Add structural HTML elements
- Style elements in CSS, one chunk at a time, from top to bottom of the page; adjust HTML as needed

Fun
CSS Tricks!

Transitions

CSS Transitions

- Lets you animate properties between states
- For us, this mainly means transitions in and out of the `:hover` state

CSS Transitions Syntax

```
a {
  transition: color 0.5s;
}
```

CSS Transitions Syntax

```
a {
  transition:
    color 0.5s,
    width 0.5s;
}
```

CSS Transitions Syntax

```
a {
  transition:
    color 0.5s ease,
    width 0.5s ease;
}
```

Transform

CSS Property: transform

- Lets you manipulate an element by skewing, rotating, moving, or scaling

```
.bigger {
  transform: scale(3);
}
```

CSS Property: transform

- `scale()` changes an element's size
- `rotate()` rotates an element
- `translate()` moves an element
- `skew()` tilts an element

CSS Property: transform

- Each requires it's own unique set of values inside the parentheses.
- Use a reference:
  - CSS Tricks Transform
  - W3 Schools Transform

Filters

CSS Property: filter

- Lets you achieve various visual effects (like Photoshop for the browser)

```
.blurry {
  filter: blur(20px);
}
```

CSS Property: filter

- `blur()` blurs an element
- `saturate()` adjusts color intensity
- `invert()` inverts color values
- `brightness()` adjusts brightness level
- `contrast()` adjusts contrast level
- `hue-rotate()` shifts color hue

Filters

CSS Property: filter

- Each requires it's own unique set of values inside the parentheses.
- Use a reference:
  - CSS Tricks Filter
  - W3 Schools Filter

Embedding
Audio & Video

Two Approaches

- Like images, we can keep audio and video assets with our web files, and use HTML tags to put them on the page:

```
<video></video>
<audio></audio>
```

Two Approaches

- Or, we can use the embed tools provided by media hosts like YouTube, Vimeo, SoundCloud, etc

Preview:
Responsive Design

What is Responsive Design?

- Responsive design means your layout functions on a continuum of devices and screen sizes
- i.e., not just a desktop + mobile site

Coding Responsiveness

- We use Media Queries to apply different CSS to different devices.
- We can apply different rules depending on screen size, device orientation (landscape or portrait), the screen's resolution

Media Query Syntax

```
@media (max-width: 480px) {
    h1 {
        width: 100%;
    }
}
```

Media Query Syntax

```
@media (min-width: 768px)
   and (orientation: landscape) {
    h1 {
        width: 768px;
    }
}
```

Media Query Resources:

- https://css-tricks.com/css-media-queries/
- https://www.w3schools.com/css/css_rwd_mediaqueries.asp

Preview:
Content Management Systems

What is a CMS?

- A tool that lets you manage (add/remove/edit) content on your site without changing the code

Popular CMS platforms:

- Blogging: WordPress, Tumblr
- Website builders: Squarespace, Wix
- Online shops: Shopify, Big Cartel

"Pure" CMS's:

- Many popular ones: Drupal, Joomla, Magento
- My fav: Craft (http://craftcms.com)

Preview:
JavaScript

What is JavaScript

- A programming language that runs in your browser that can:
  - Listen for user interaction
  - Manipulate any element on the page

Where to put JavaScript

- Simple scripts can go in your HTML inside a script tag:

```
<script>
    alert("Hello, I am some JavaScript!");
</script>
```

What is JQuery?

- A popular JavaScript library that simplifies locating and manipulating elements.

What else can JavaScript do?

- 3D!
- Games! Tools! Art! (really anything inside the browser)
- Things outside the browser too! Desktop applications, tablet/mobile apps, server-side applications

Preview:
Frameworks & Libraries

What is a Framework/ Library?

- A bunch of code meant to help you solve common web development problems and write more complex applications

Bootstrap

http://getbootstrap.com

- HTML, CSS, and JS library with a bunch of nicely designed, reusable components
- A CSS grid system that makes it easy to create responsive layouts

Frameworks & Libraries

D3.js

http://d3js.org

- JavaScript library that lets you create powerful data visualizations

React

http://reactjs.org

- Library that builds out your HTML for you using JavaScript
- Makes it easier to make applications where the page changes frequently

Paths Ahead:
Apps, Games, & Artwork

Paths Ahead

Apps

Encompasses all manner of software solving all kinds of problems—tools to let people create & learn, interact with each other, organize information, etc.

Games

Create enjoyment, tell rich stories, build an emotional connection with a player.

Recommended resources:
Udemy course
PICO-8

Paths Ahead

Artwork

Amazing, underutilized way to learn coding and explore coding ideas more intimately.

Recommended resources:
http://p5js.org
http://thecodingtrain.com

Class Projects!

That's all folks!

Please provide feedback for this class:

http://svcseattle.com/evaluations

& Keep in touch!

scott@movingobjects.io