

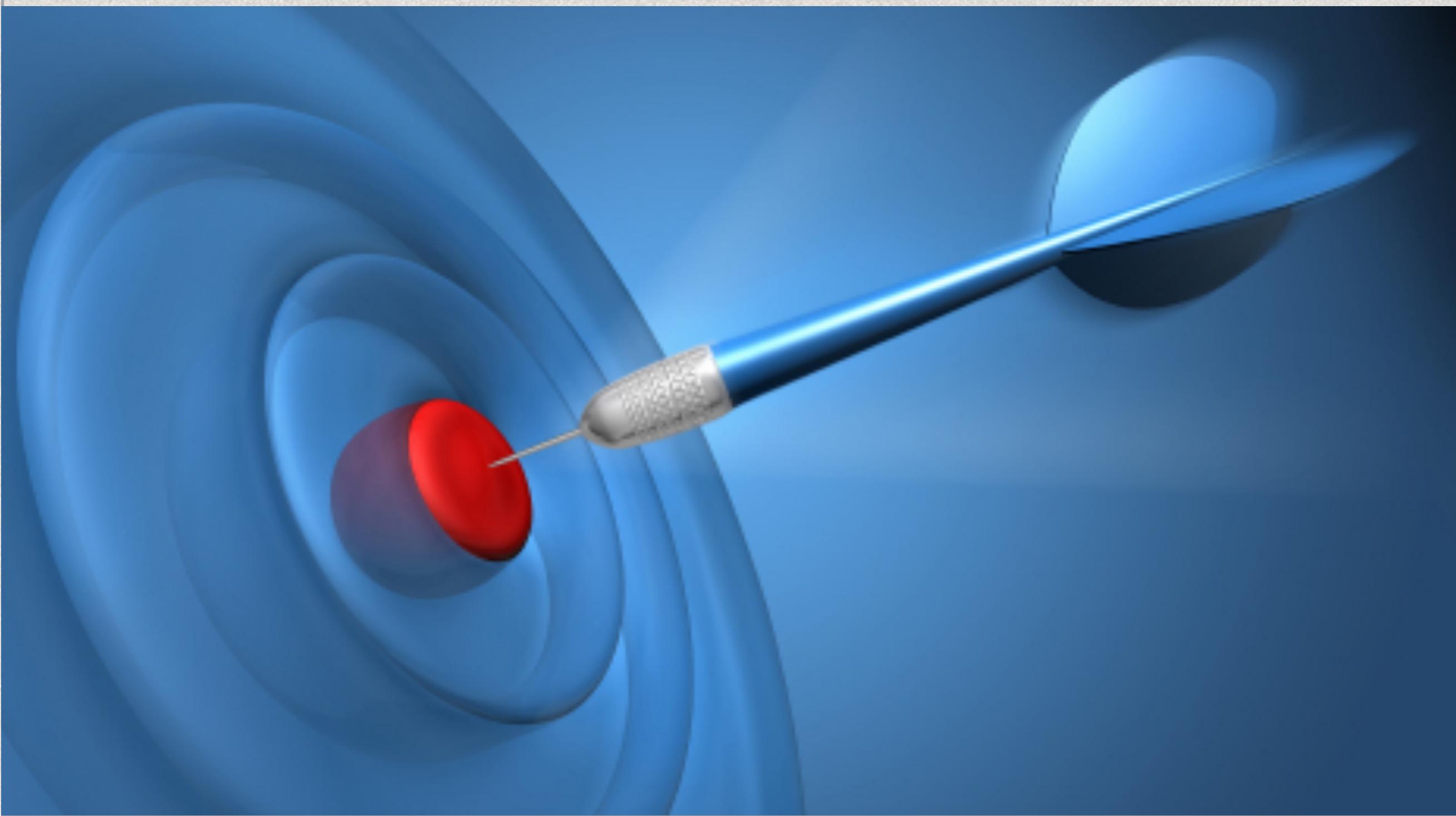
Pro CSS

Designing With Style

Agenda

- * CSS Selectors
- * CSS Properties
- * Page Layouts
- * Tips & Tricks

CSS Selectors



Class Selector

Selects all elements having a class attribute

CSS

```
img.thumb { width: 32px; height: 32px; }
```

HTML

```
<div class="gallery">
  <ul>
    <li></li>
    <li></li>
    <li></li>
  </ul>
</div>
```

ID Selector

Select a node based on id

CSS

```
#content { width: 960px; margin: 0 auto; }
```

HTML

```
<div id="content">  
</div>
```

Descendants Selector

Select all descendants of the given element

CSS

```
.gallery img { border: 2px solid blue; }
```

HTML

```
<div class="gallery">
  <ul>
    <li></li>
    <li></li>
    <li></li>
  </ul>
</div>
```

Child Selector

Select a direct child.

CSS2.1 Selector. IE8 and above

CSS

```
.header > h1 { font-size: 1.8em; font-weight: bold; }
```

HTML

```
<div class="header">
  <h1>Welcome Home</h1>
  <div class="menu">
    <h1>About</h1>
    <h1>Gallery</h1>
    <h1>Contact</h1>
  </div>
</div>
```

Attribute Selector

Select an element with a specific attribute
CSS2.1 Selector. IE7 And above

CSS

```
input[type="text"] { border: 2px solid blue; }
```

HTML

```
<form>
  <input type="text" name="name" />
  <input type="text" name="phone" />
  <input type="submit" value="Go" />
</form>
```

Enter CSS3

- ✳ More selectors
- ✳ Less clutter in HTML



Advanced Attribute

Select an attribute that “starts with”
IE7 and above

css

```
a[href^="mailto:"] {  
    background: url(/images/icn-email.png);  
}
```

HTML

```
<a href="/work">Work</a>  
<a href="/about">About</a>  
<a href="mailto:admin@work.com">Contact</a>
```

Advanced Attribute

Select an attribute that “ends with”
IE7 and above

CSS

```
a[href$="pdf"] {  
    background: url(/images/icn-pdf.png);  
}
```

HTML

```
<a href="report.pdf">Report</a>
```

Pseudo Selectors

- * :first-child
- * :last-child
- * :nth-child, :nth-of-type
- * :not
- * :enabled, :disabled, :checked

:nth-child

```
td { padding: 2px; }
tr td:first-child { width: 100px; }
tr:nth-child(2n) { background: #ccc; }
```

```
<table>
  <tr>
    <td>Dollar</td>
    <td>USA</td>
  </tr>
  <tr>
    <td>Pound</td>
    <td>Great Britain</td>
  </tr>
  <tr>
    <td>Yen</td>
    <td>Japan</td>
  </tr>
  <tr>
    <td>Euro</td>
    <td>EMU</td>
  </tr>
</table>
```

Dollar	USA
Pound	Great Britain
Yen	Japan

:nth-child

- * An element that has **an+b-1 siblings before in the document tree**
- * Very useful for zebra tables
- * Can save html classes: first, second, third, etc.

:nth-of-type

- * Has **an+b-1 siblings with the same expanded element name before**
- * Used for mixed type containers

Psudo Elements

- * Create abstractions about the document tree
beyond html
- * Goal: Simplify your html
- * ::first-letter, ::first-line, ::before, ::after

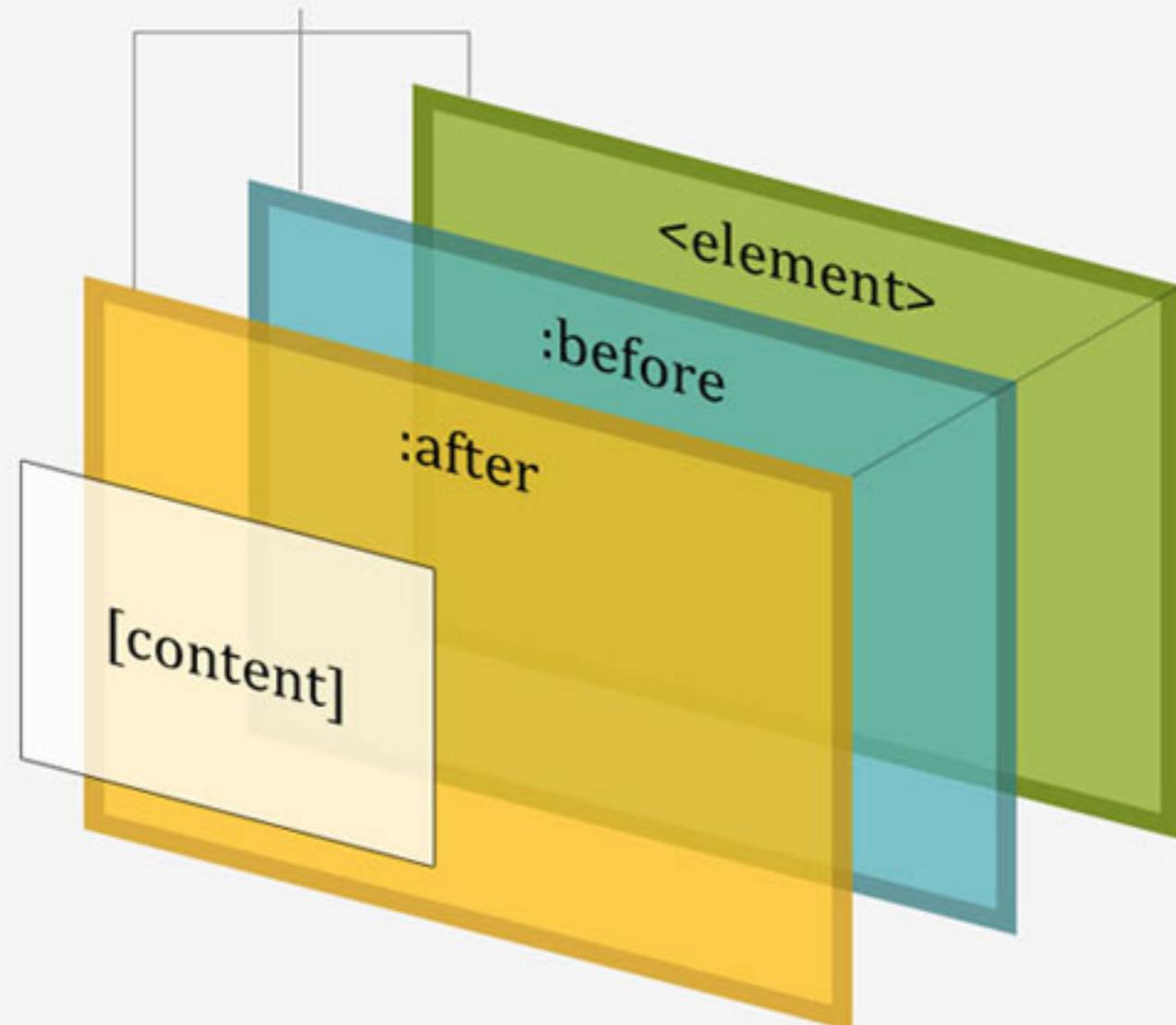
Drop Caps

```
p { margin-top: 20px; }
p::first-letter {
    font: 2em zapfino;
    float: left;
    margin: -22px 15px 0 0;
}
```

P ellentesque habitant morbi tristique senectus et
netus et malesuada fames ac turpis egestas.
Vestibulum tortor quam, feugiat vitae, ultricies
egestas semper. Aenean ultricies mi vitae est. Mauris
placerat eleifend leo.

:before and :after

Multiple boxes from one HTML element allow for multiple backgrounds and borders



Demos:

- * Intro: <http://jsfiddle.net/kcYUM/1/>
- * Shapes: <http://css-tricks.com/examples/ShapesOfCSS/>
- * Multiple borders: <http://nicolasgallagher.com/multiple-backgrounds-and-borders-with-css2/demo/borders.html>
- * Ribbons: <http://jsfiddle.net/y5u3k/>

Lab

- * Write a CSS selector for each marked element in the following HTML.
- * Test your work by changing background color for the selected element
- * <http://jsfiddle.net/9GYsp/>

All 3 A's together

```
<body>
  <nav>
    <ul>
      <li><a href="#">Home</a></li>
      <li><a href="#">About</a></li>
      <li><a href="#">Schedule</a></li>
    </ul>
  </nav>

  <div id="page-main">
    <article>
      <p>Pellentesque habitant mor.</p>
      <ul>
        <li>Lorem ipsum .</li>
        <li>Aliquam tin.</li>
        <li>Vestibulum .</li>
      </ul>
      <p>Pellentesque habitant mor.</p>
    </article>
  </div>
</body>
```

CSS Conflicts



phillipmartin.info

What Happens Here ?

css

```
.foo { color: blue; }  
p     { color: red; }
```

HTML

```
<p class="foo">Hello</p>
```

What Happens Here ?

css

```
p { color: red; }  
.foo { color: blue; }
```

HTML

```
<p class="foo">Hello</p>
```

RESULT

```
Hello
```

Selector Specificity

- * Number of IDs = A
- * Number of class, attributes, pseudo-classes = B
- * Number of type, pseudo elements = C
- * Add it all to get the specificity
- * Selector with the larger specificity wins

CSS Faceoff

Who Wins ?

#slider

OR

div.twocol .container .primary .content ul

CSS Faceoff

- * .container .external <=> .container a[rel="external"]
- * div ul li a.item <=> div.nav li a

IE < 9 Shim

- * Most interesting selectors require a modern browser
- * If your clients don't have one - use a shim

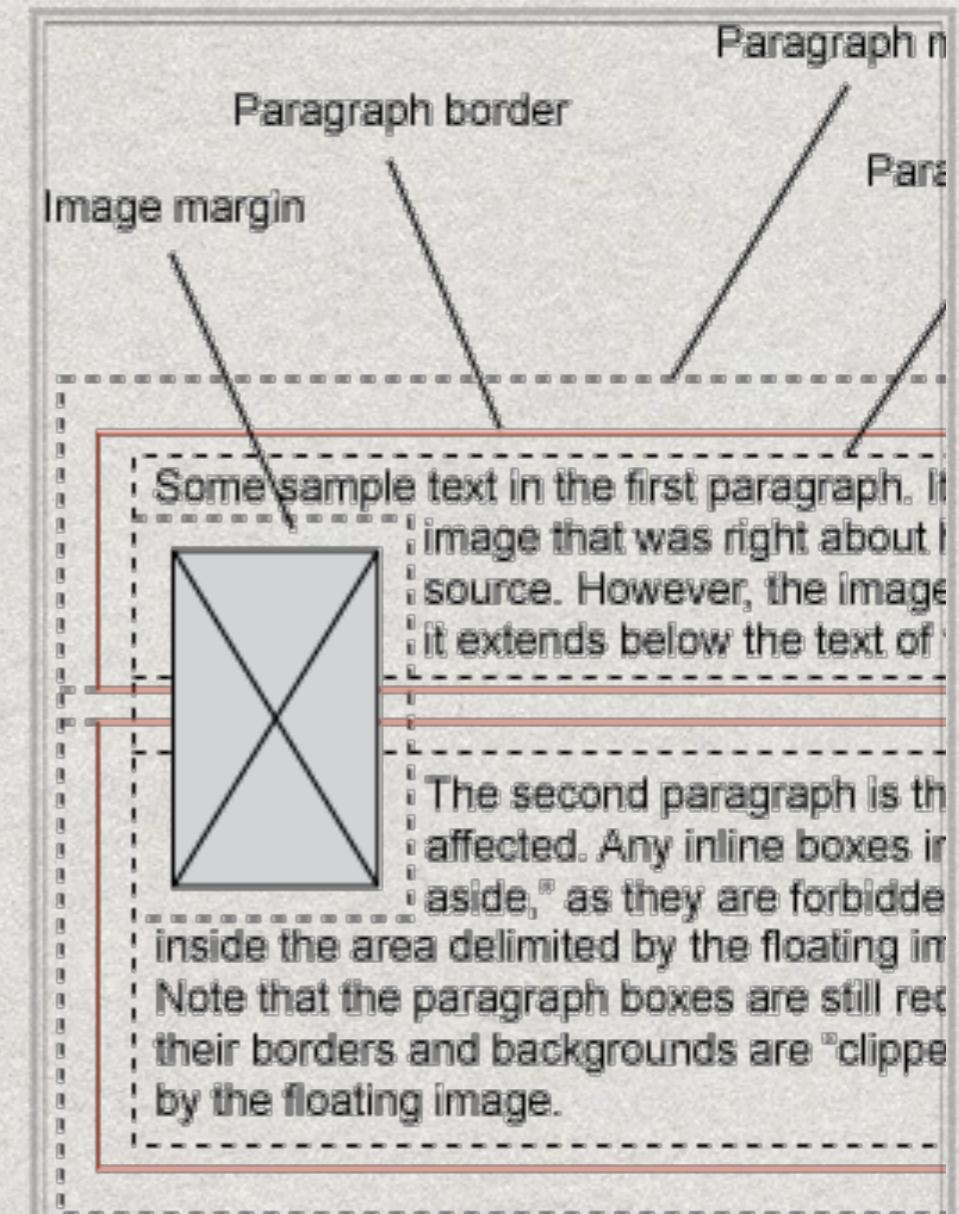
:select[ivizr]

CSS Properties

- * float and positioning
- * display: block, inline, inline-block
- * background

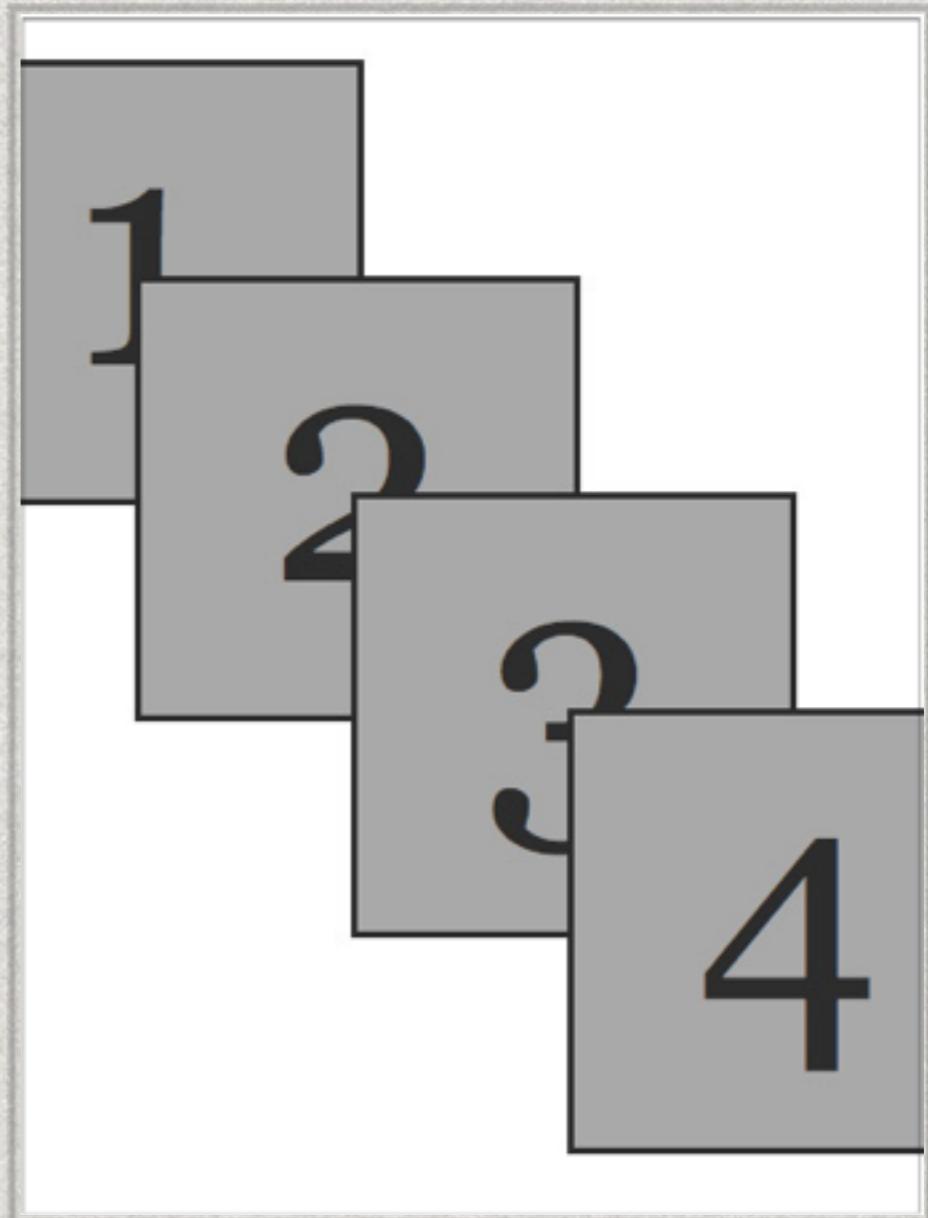
FLOATS

- * Takes an element out of the normal layout context
- * “wrap” all other contents around it



position: absolute

- * Draw an element in a specific location.
- * The element is painted in another “layer”
- * Positioning is relative to a **container having non static positioning**



Display: inline/block

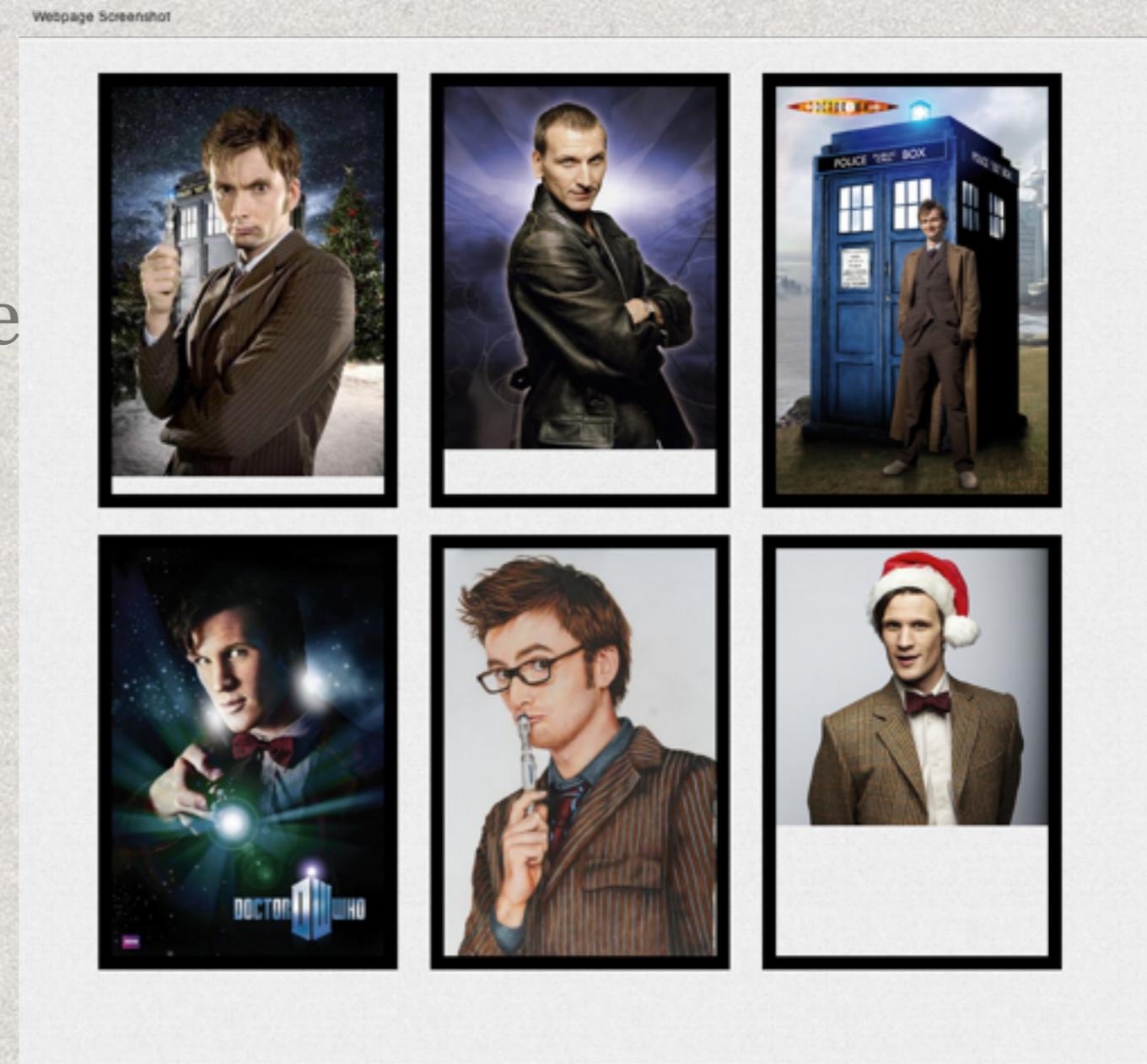
- * Block elements occupy an entire line
- * Inline elements are placed inside box elements
- * Common block elements: div, h1..h6, p, ul, ol, dl, li, dt, dd, table, blockquote, pre, form
- * Common inline elements: span, a, strong, em, img, br, input
- * inline-block combines the two

Demo

- * Implementing a top navigation menu with inline-block

Lab

- * Fill in the CSS to create the grid on the right
- * HTML at:
<https://gist.github.com/4286294>



Page Layouts

- * Fixed layout
- * Fluid (liquid) layout
- * Adaptive layout

Fixed Width Layout

- * Usually 960px or 760px wide
- * Keep your designers happy
- * Example: <http://www.petplace.com/>

Fluid Width Layouts

- * Use percentages for widths
- * Can make better use of screen real estate than fixed
- * Use min-width, max-width to prevent surprises
- * Remember: **Page must be usable at all times**
- * Example: <http://graybit.com/main.php>

Adaptive Layout

- * Uses media queries to make a smart decision about the layout
- * Makes the best use of screen real estate
- * Takes the most work to implement
- * Example: mediaqueri.es

Choosing Layout

- * Be careful with **fluid** layouts - easiest to build, most difficult to test
- * Use **fixed** layout when targeting **only one** client
- * Use **adaptive** layout when targeting **many** client platforms

960 Grid

- * For fixed grid layouts, no need to reinvent the wheel.
- * 960grid is a set of CSS files for implementing a grid system

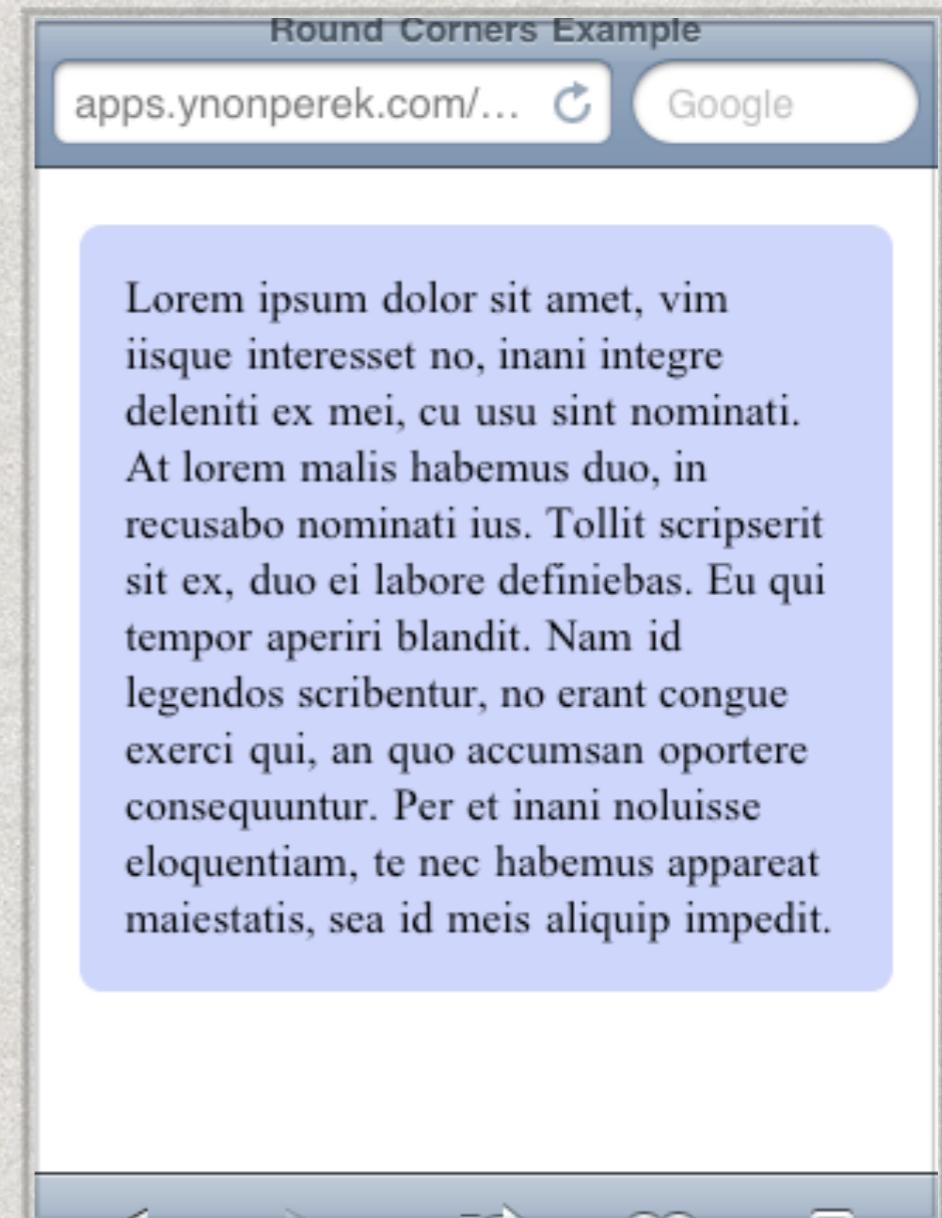
CSS3 New Features

- * Rounded Corners
- * Shadows
- * Gradients
- * Transitions & Animations

Rounded Corners

- * use **-prefix-border-radius** to get the effect
- * No need to use background images
- * Sample use:

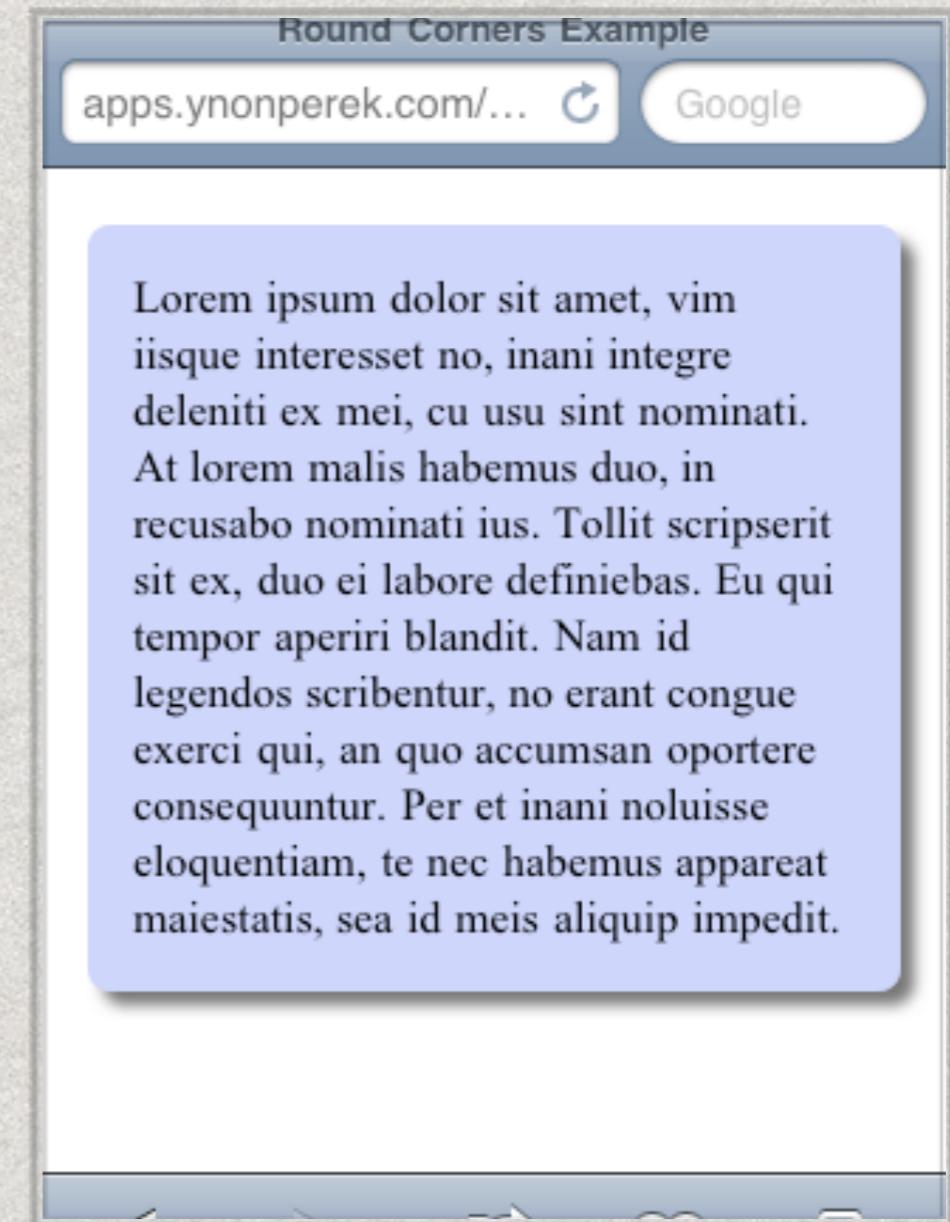
`-webkit-border-radius: 8px;`



Shadows

- * use **-prefix-box-shadow** to get a shadow effect
- * `rgba(0, 0, 0, 0.6)` will work as the shadow color
- * Sample Use:

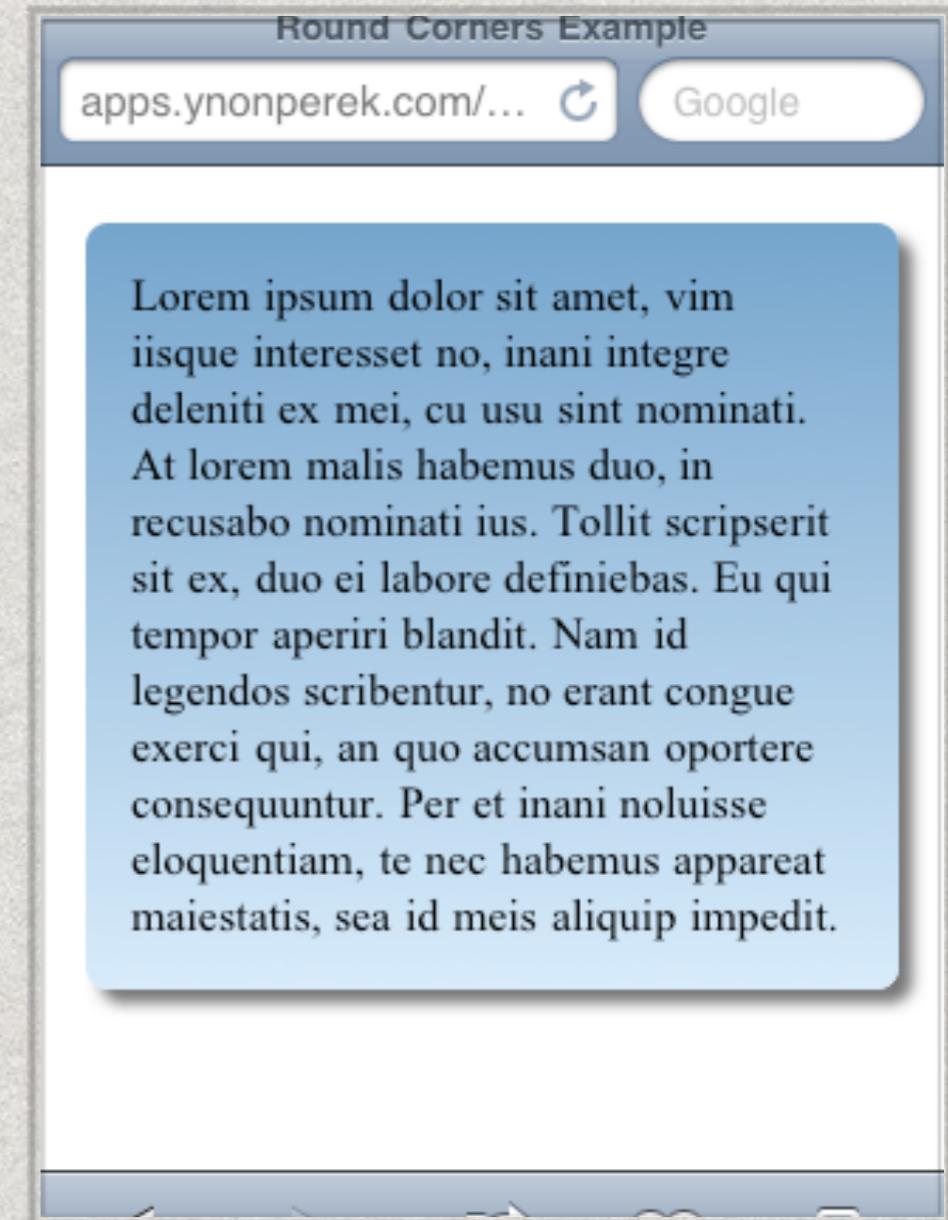
```
-webkit-box-shadow: 5px 5px 5px  
rgba(0, 0, 0, 0.6);
```



Gradients

- * CSS 3.0 added support for gradients
- * use them instead of background images to save bandwidth
- * Online gradient generator:

<http://www.colorzilla.com/gradient-editor/>



CSS Transition

- * Controls the transition between element states
- * Allows animating transitions
- * Uses 3D acceleration

Demo - Transitions

- * Each state is represented by a CSS class
- * Clicking the button changes element state



-webkit-transition

- * transition-property
- * transition-duration
- * transition-timing-function
- * transition-delay



transition-property

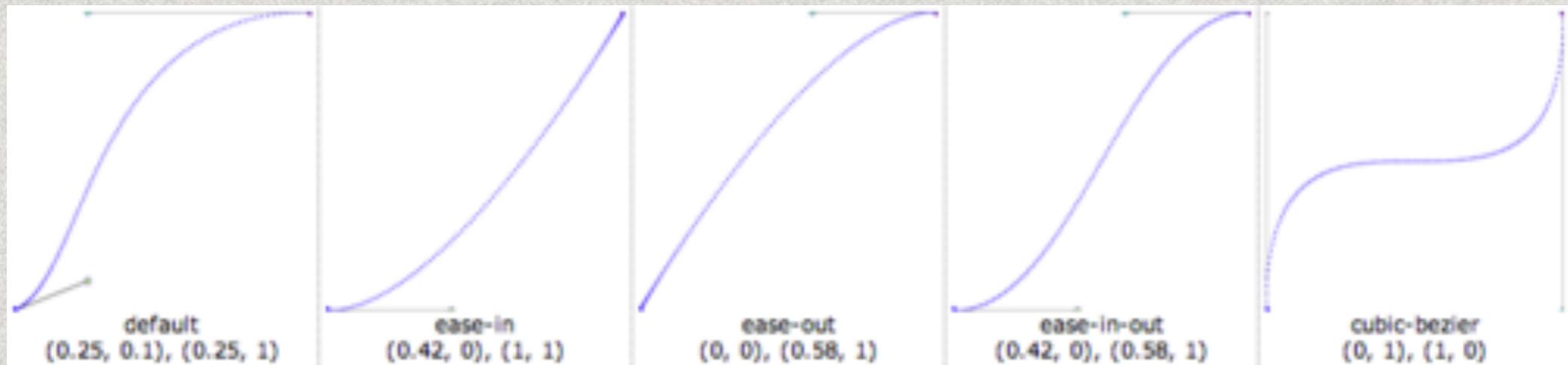
- ✳ Almost any property can be animated
- ✳ Use all to catch everything
- ✳ Full list at: [https://developer.mozilla.org/en/css/css transitions](https://developer.mozilla.org/en/css/css_transitions)

transition-duration

- ✳ How long the property animation should take
- ✳ Use time in seconds or ms (can be < 1s)

transition-timing-function

- * Determines the timing function for the animation



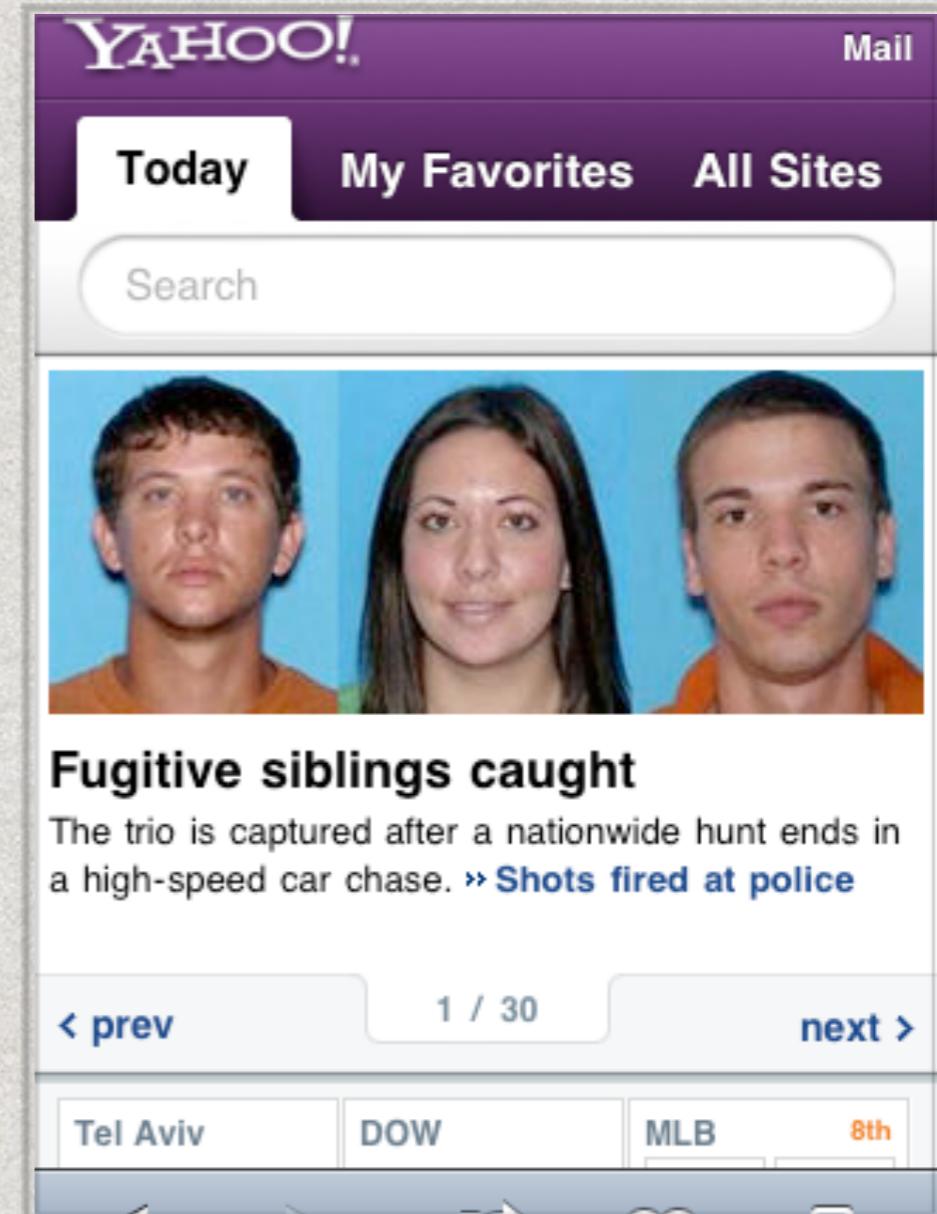
Live demo at: <http://www.the-art-of-web.com/css/timing-function/>

transition delay

- * Delay between value change and animation start
- * Can be used to coordinate multiple animating objects

Exercise

- * Implement a yahoo style “top news” section
- * Click next to animate to the next image
- * Click prev to animate to the previous image
- * Hint: 3 divs inside a container, and animate position

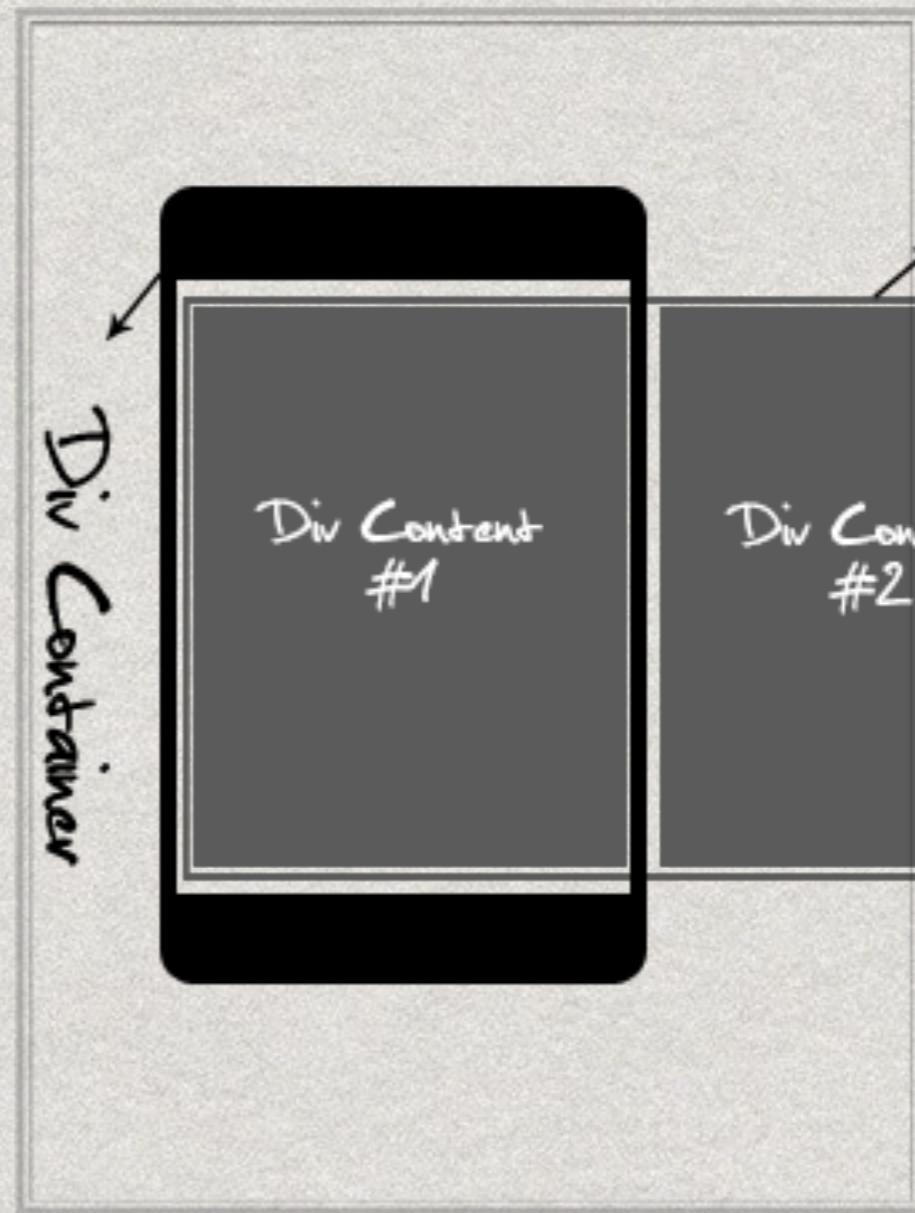


Transition Effects

- * Mobile apps usually have some native animations for changing screens
- * On the web, we can implement these using css transformations

Slide Effect

- * A slide effect is built of two child divs and a parent with overflow:hidden
- * Sliding is achieved by changing the translate-x of the child divs



CSS Transformations

```
-webkit-transform:  
    rotate(25deg)  
    translate(80px, 150px);
```

All Your Base Are Belong To Us

CSS Transformations

- * Available Transformations:
 - * translate
 - * scale
 - * rotate
 - * skew

Keyframe Animations

- * Use keyframe animations when:
 - * Need more control
 - * Repeated Animations

Keyframe Animations

- * Define animation
- * Glue it to an element

Keyframe Animations

- * Define An Animation With:

```
@-webkit-keyframes rounder {  
  0%   { border-radius: 0; }  
  100% { border-radius: 100px; }  
}
```

Keyframe Animations

- * Glue it to an element

```
div {  
    width: 200px;  
    height: 200px;  
    background: blue;  
    -webkit-animation: rounder 2s infinite;  
}
```

Keyframes Lab

- * Write a spinner keyframe animation



Common CSS

- * Centering elements
- * Showing/Hiding elements
- * Box punching
- * CSS Sprites



Q & A

Thank You

- * Ynon Perek
- * ynon@ynonperek.com
- * ynonperek.com