CSS Overview

Selectors, Integration, Inheritance, Cascading

R. Scott Granneman & Jans Carton

2.4

Notes & URLs for this presentation can be found...

» underneath the link to this slide show on granneman.com
» at files.granneman.com/presentations/webdev/CSS-Overview.txt

Scott Granneman

Interviews

Ladue Chapel

Exploring HTML5 & CSS

Advanced CSS

HTML to CSS to Responsive Web Design

Responsive Web Design for Designers

Web Design

Archived Presentations

Creative Commons License

Scott Grann... in a Web development company in St. Louis, Missouri. Scott has written six books, contributed ... r SecurityFocus and *Linux Magazine*. He is an Adjunct Professor at Washington University in St. Louis ... n technology, security, web development, & the Internet. As a Principal of WebSanity, he plans, devel ... states using an open source, UNIX-based content management system. He has given presentatio ... peaking to people.

Key S

Pages you'r...
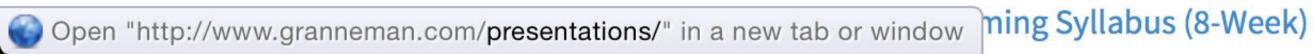
Personal

- Publications
- Summer Vacation 2007: The Plains

Teaching

- Web Development Key Links
- Wash U: Website Design & Development Syllabus
- Webster U (Comp Sci): Intro to Web Programming Syllabus
- ...ming Syllabus (8-Week)

Open "http://www.granneman.com/presentations/" in a new tab or window

Gary Bernhardt
@garybernhardt

Watching me do CSS is like watching a drunk toddler in zero gravity repair a watch with mittens on. Also the watch randomly catches fire.

RETWEETS
59

LIKES
99

12:36 PM - 10 Nov 2015

History

CSS 1: December 1996

CSS 2: May 1998

CSS 2.1: July 2007

CSS 3: June 1999–Now

CSS 4: 2012–Now

CSS 1 & 2 each were one big document

CSS 3 (& 4) is not one large single spec

Instead, it's divided into many separate documents called *modules*

Currently 50+ modules!

Different modules have different statuses

www.w3.org/Style/CSS/current-work

CSS Grid Layout 📄 - CR

Global | 69.53% + 5.62% = | 75.15%

unprefixed: | 69.53%

Method of using a grid concept to lay out content, providing a mechanism for authors to divide available space for layout into columns and rows using a set of predictable sizing behaviors

Current aligned | Usage relative | Date relative | Show all

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
			[1] 49 🚩			10.2			
	[2] 15 ⊟	55	60	10.1	47	10.3		4.4	
[2] 11 ⊟	16	56	61	11	48	11	all	56	61
		57	62	TP	49				
		58	63		50				
		59	64						

Notes | Known issues (2) | Resources (12) | Feedback

W3C indicates the maturity of specifications by a status code. The CSS working group uses the following, from *least* to *most stable:*

Abbreviation	Full name
FPWD	First Public Working Draft
WD	Working Draft
CR	Candidate Recommendation
PR	Proposed Recommendation
REC	Recommendation
SPSD	Superseded Recommendation

The following code indicates a document that is not intended to become a standard:

Abbreviation	Full name
NOTE	Working Group Note

The names are defined in section 6 of the W3C process document. A REC is what is normally referred to as a 'standard.' W3C encourages everyday use starting from CR.

W3C CSS Working Group: Colors & Status Codes

Why CSS?

Separation of Concerns

Divide computer program into separate *concerns*, each focusing on a specific resource

Meaning: HTML

Presentation: CSS

Behavior: JavaScript

Separate content (HTML) from presentation (CSS)

Site-wide consistency: control how all content looks using only 1 (or a few) CSS file/s

Apply different styles to same content in different *media*:

» desktop web browser
» mobile web browser
» auditory
» print
» & more!

Adherence to standards

🎉 🎊 🥳 🤼‍♂️ 🕺

It's fun!

Integrating CSS

4 ways to connect your HTML with your CSS

1. Inline styles
2. Embedded styles
3. Linking to external styles

4. `@import`

Inline

Uses the `style` global attribute

```html
<p style="color: red; font-weight: bold;
font-size: 180%;">
  Do not disturb Cthulhu!
</p>

<p>
  Enjoy your trip to R'yleh.
</p>

<p>
  We hope you emerge relatively sane.
</p>
```

HTML

Do not disturb Cthulhu!

Enjoy your trip to R'yleh.

We hope you emerge relatively sane.

CSS

JS

Quick & easy to create, but difficult & time-consuming to manage

Must repeat over and over

Can't change the style according to the media, so styles apply to all media

Doesn't separate content & presentation

So why use it?

~~Testing~~ (use the Inspector instead)

Use inline styles for unique instances (*very* rare!)

High specificity: overrides conflicting declaration (only when absolutely necessary!)

JavaScript often uses inline styles to apply styles dynamically

HTML email

Embedded

Styles inserted inside `<style>` … `</style>`

Most often in the `<head>`, but can be found anywhere in the `<body>`

```html
1  <head>
2    <meta charset="UTF-8">
3    <title>Embedded Styles
   Example</title>
4    <style>
5      html {
6        font-size: 24px;
7        font-family: "Iowan Old
   Style", Georgia, serif;
8      }
9      h1, h2 {
10       font-family: sans-serif;
11     }
12   </style>
13  </head>
14
15  <body>
16    <h1>Cthulhu</h1>
17    <p>Ph'nglui mglw'nafh Cthulhu
   R'lyeh wgah'nagl fhtagn!</p>
18    <h2>Nyarlathotep</h2>
19    <p>Ph'nglui mglw'nafh Cthulhu
   R'lyeh wgah'nagl fhtagn!</p>
20  </body>
```

Cthulhu

Ph'nglui mglw'nafh Cthulhu R'lyeh wgah'nagl fhtagn!

Nyarlathotep

Ph'nglui mglw'nafh Cthulhu R'lyeh wgah'nagl fhtagn!

Embedded styles are great for one page ...

... but they rapidly become difficult to manage on multiple pages

So why use it?

Resource inlining: embedding reduces outbound requests (so does inline CSS)
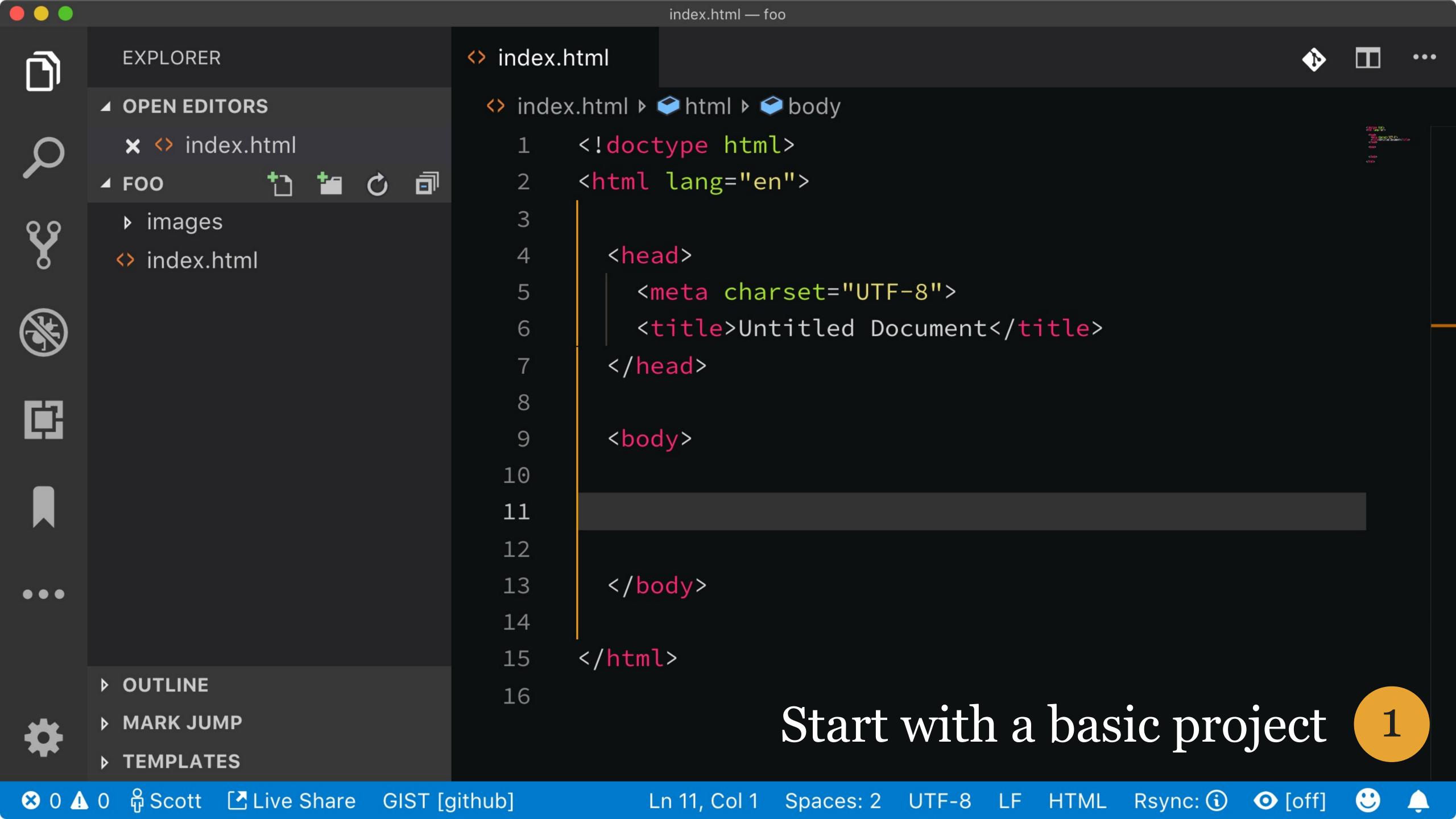
Portability: You have a widget that may get embedded into another webpage on another site
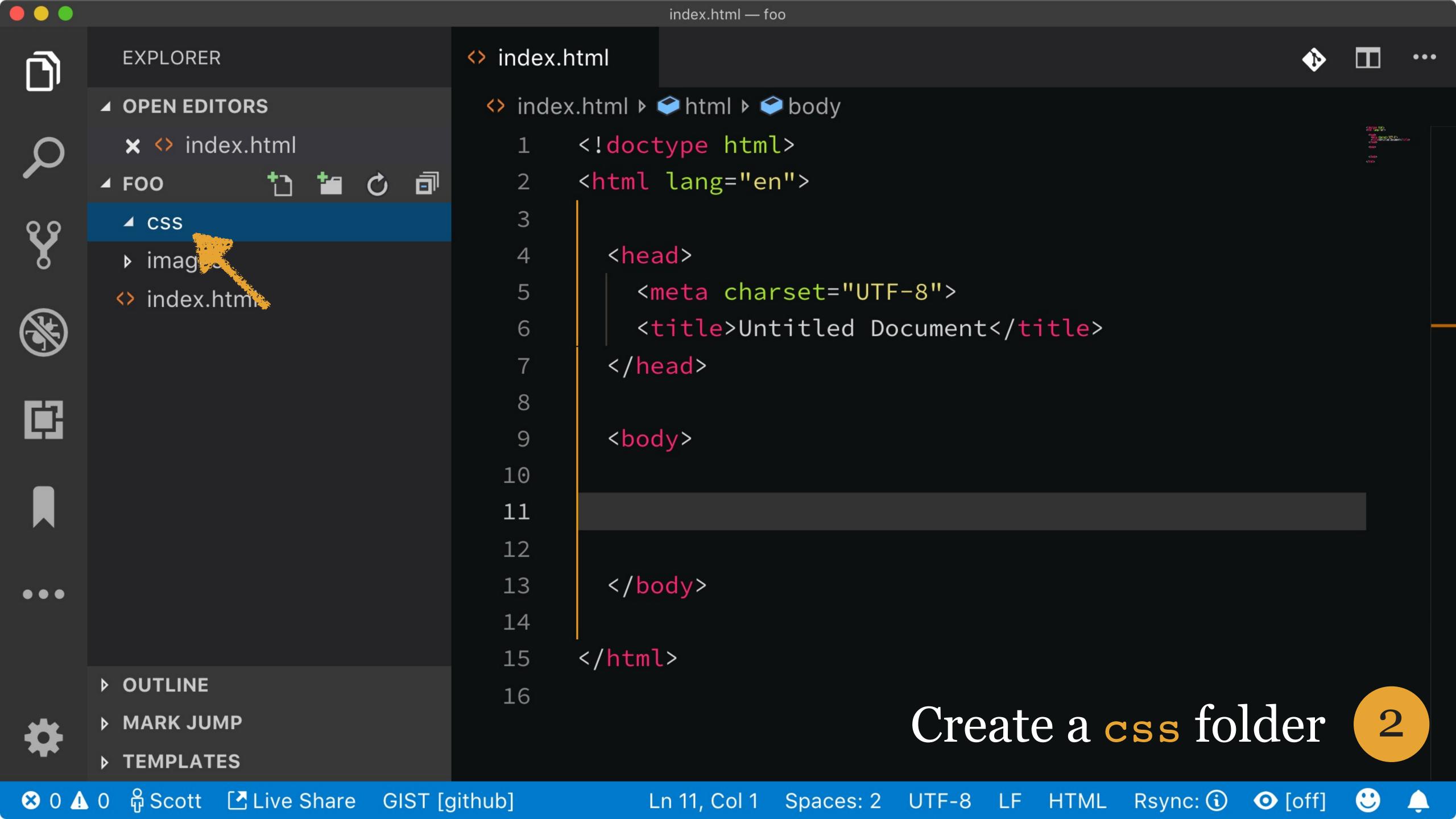
To style the widget, you include embedded styles with it inside `<style>` ... `</style>`

Linking

HTML 4.01
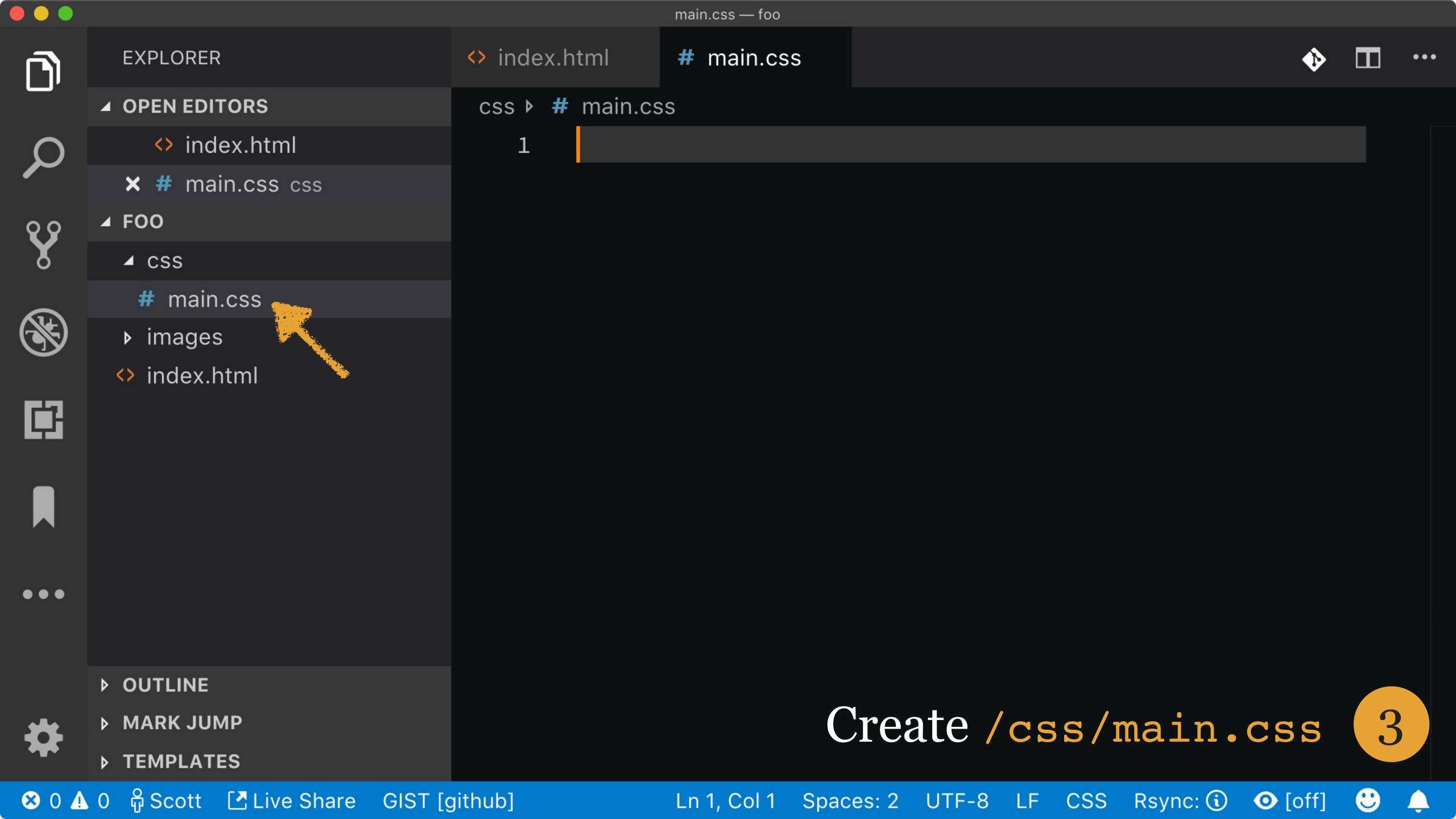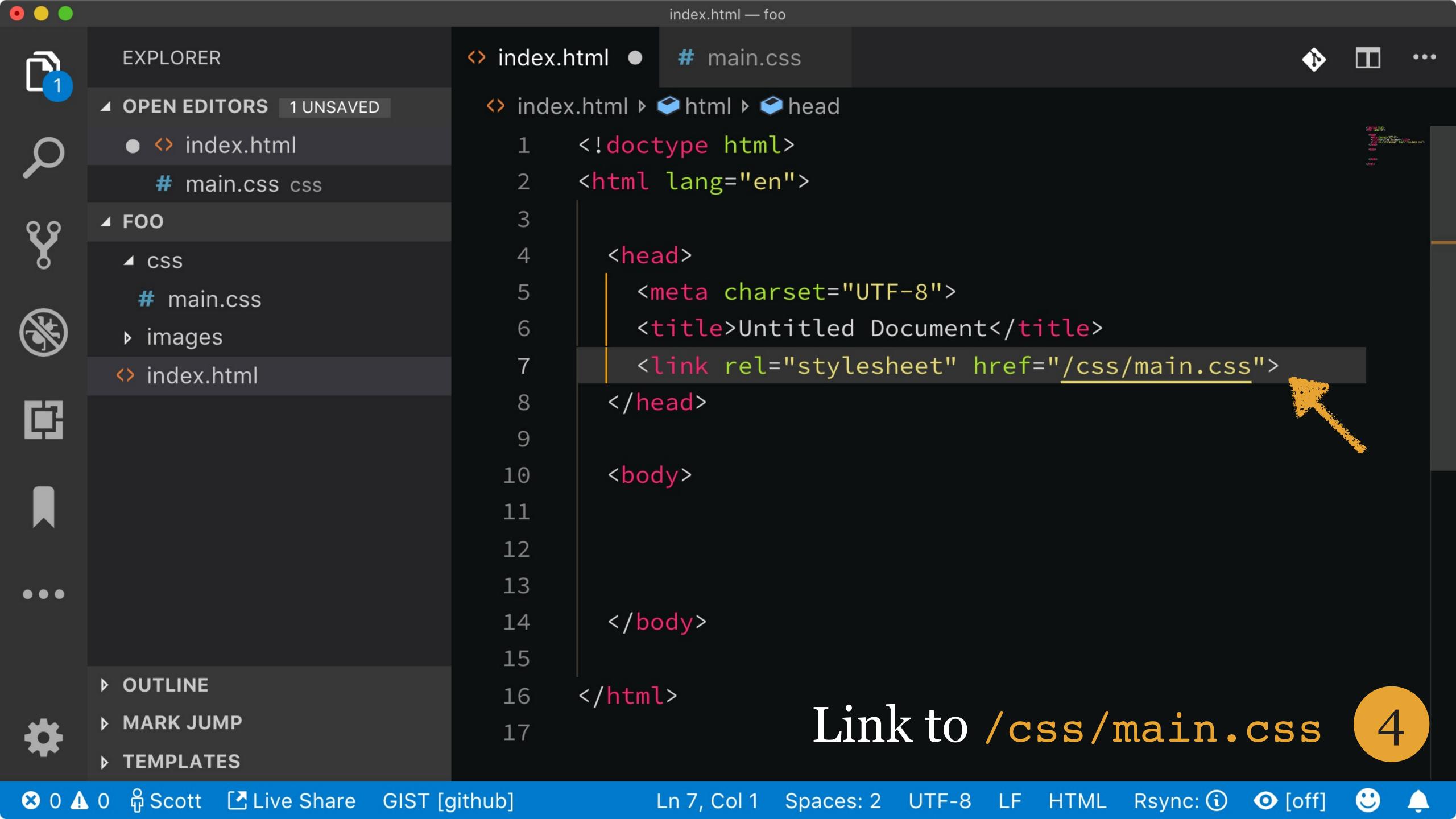
```html
<head>
  <link rel="stylesheet" type="text/css"
  href="/css/main.css">
</head>
```

HTML 5

```html
<head>
  <link rel="stylesheet" href="/css/main.css">
</head>
```

EXPLORER

FOO

▸ images
<> index.html

<> index.html

<> index.html ▸ 🔷 html ▸ 🔷 body

```
 1   <!doctype html>
 2   <html lang="en">
 3
 4     <head>
 5       <meta charset="UTF-8">
 6       <title>Untitled Document</title>
 7     </head>
 8
 9     <body>
10
11
12
13     </body>
14
15   </html>
16
```

▸ OUTLINE

▸ MARK JUMP

▸ TEMPLATES

Start with a basic project 1

🛇 0 ⚠ 0 👤 Scott ⎘ Live Share GIST [github]   Ln 11, Col 1   Spaces: 2   UTF-8   LF   HTML   Rsync: ⓘ 👁 [off] 🙂 🔔

EXPLORER

<> index.html

OPEN EDITORS

✕ <> index.html

FOO

▲ css

▶ images

<> index.html

OUTLINE

MARK JUMP

TEMPLATES

<> index.html ▸ 🔷 html ▸ 🔷 body

```html
1    <!doctype html>
2    <html lang="en">
3
4      <head>
5        <meta charset="UTF-8">
6        <title>Untitled Document</title>
7      </head>
8
9      <body>
10
11
12
13      </body>
14
15   </html>
16
```

Create a `css` folder 2

EXPLORER

<> index.html   # main.css

css ▸ # main.css

1

OPEN EDITORS

<> index.html

✕ # main.css css

FOO

◢ css

main.css

▸ images

<> index.html

OUTLINE

MARK JUMP

TEMPLATES

Create /css/main.css ③

⊗ 0 ⚠ 0   Scott   Live Share   GIST [github]   Ln 1, Col 1   Spaces: 2   UTF-8   LF   CSS   Rsync: ⓘ   [off]

EXPLORER

<> index.html ●   # main.css

OPEN EDITORS   1 UNSAVED
● <> index.html
main.css css

FOO
▲ css
main.css
▸ images
<> index.html

<> index.html ▸ ⬡ html ▸ ⬡ head

```html
1   <!doctype html>
2   <html lang="en">
3
4     <head>
5       <meta charset="UTF-8">
6       <title>Untitled Document</title>
7       <link rel="stylesheet" href="/css/main.css">
8     </head>
9
10    <body>
11
12
13
14    </body>
15
16  </html>
17
```

▸ OUTLINE
▸ MARK JUMP
▸ TEMPLATES

Link to /css/main.css ④

⊗ 0 ⚠ 0   Scott   Live Share   GIST [github]   Ln 7, Col 1   Spaces: 2   UTF-8   LF   HTML   Rsync: ⓘ   ⊙ [off]

What should you name your CSS file?

It doesn't matter

```
main.css
typography.css
client.css
search.css
navigation.css
```

Where should you place your CSS file?

In your website's root directory, always create these directories:

`css` (or `styles`)
`fonts`
`images` (or `media`)
`js` (or `scripts`)

A CSS file is solely made up of rulesets & comments

```css
/* Common */
blockquote, p, td {
  font-family: Verdana, sans-serif;
  font-size: 1em;
}


#footer {
  font-size: .9em;
}


.emphasis {
  font-weight: bold;
}
```

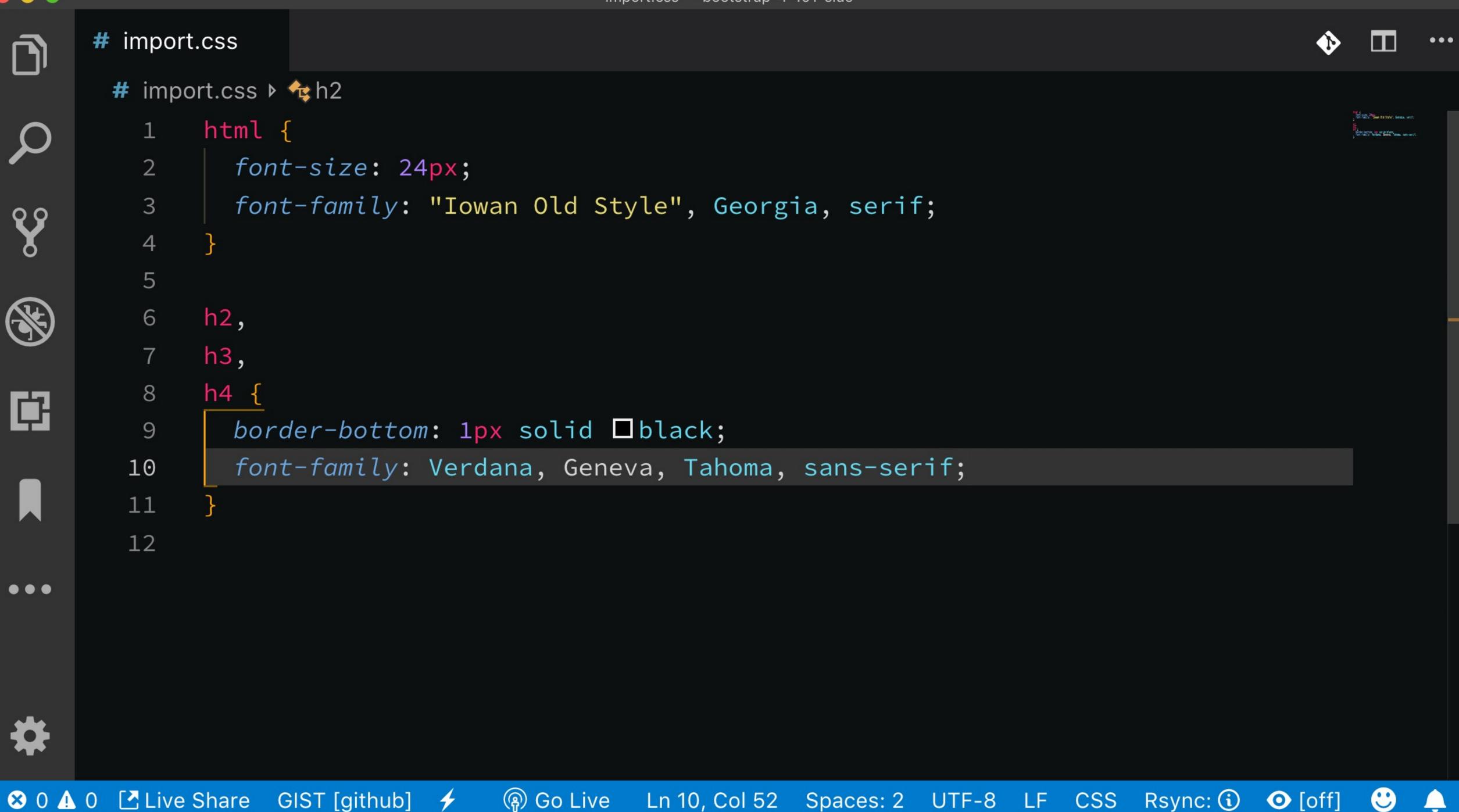You can link to more than one style sheet, but you should try to keep those links to a minimum

If you have more than one webpage, you really ought to use an external style sheet

You can now change the look & behavior of an entire site by changing only one document!

@import

`@import` allows you to include external stylesheets in your CSS; in other words, it allows you to link to another CSS file from within a CSS file (yes, this is a little weird)

`@import` must always come 1st, ahead of any other CSS

import.css

import.css ▸ h2

```css
1   html {
2       font-size: 24px;
3       font-family: "Iowan Old Style", Georgia, serif;
4   }
5
6   h2,
7   h3,
8   h4 {
9       border-bottom: 1px solid black;
10      font-family: Verdana, Geneva, Tahoma, sans-serif;
11  }
12
```

```html
<h2>The Shadow out of Time
<small>H.P.
Lovecraft</small></h2>
<p>
    After twenty-two years of
nightmare and terror, saved only
by a desperate conviction of the
mythical source of certain
impressions, I am unwilling to
vouch for the truth of that which
I think I found in Western
Australia on the night of July
17-18, 1935. There is reason to
hope that my experience was
wholly or partly an
hallucination-for which, indeed,
abundant causes existed. And yet,
its realism was so hideous that I
sometimes find hope impossible.
If the thing did happen, then man
```

CSS

```css
@import url("https://s3-us-
west-2.amazonaws.com/s.cdpn.io
/122116/import.css");
```

JS

The Shadow out of Time H.P. Lovecraft

After twenty-two years of nightmare and terror, saved only by a desperate conviction of the mythical source of certain impressions, I am unwilling to vouch for the truth of that which I think I found in Western Australia on the night of July 17–18, 1935. There is reason to hope that my experience was wholly or partly an hallucination—for which, indeed, abundant causes existed. And yet, its realism was so hideous that I sometimes find hope impossible. If the thing did happen, then man must be prepared to accept notions of the cosmos, and of his own place in the seething vortex of time, whose merest mention is paralysing. He must, too, be placed on guard against a specific lurking peril which, though it will never engulf the whole race, may impose monstrous and unguessable horrors upon certain venturesome members of it. It is for this latter reason that I urge, with all the force of my being, a final abandonment of all attempts at unearthing those fragments of unknown, primordial masonry which my expedition set out to investigate.

import.css included via @import

HTML

```
1  <h2>The Shadow out of Time
   <small>H.P.
   Lovecraft</small></h2>
2  <p>
3    After twenty-two years of
   nightmare and terror, saved only
   by a desperate conviction of the
```

CSS

```
1  @import url("https://s3-us-
   west-2.amazonaws.com/s.cdpn.io
   /122116/import.css");
2
3  h2, h3, h4 {
4    font-family: -apple-system,
   BlinkMacSystemFont, "Segoe UI",
   Roboto, Oxygen, Ubuntu,
   Cantarell, "Fira Sans", "Droid
   Sans", "Helvetica Neue", Arial,
   sans-serif;
5  }
6
7  small {
8    display: block;
9    margin-top: .25em;
10 }
```

JS

The Shadow out of Time
H.P. Lovecraft

After twenty-two years of nightmare and terror, saved only by a desperate conviction of the mythical source of certain impressions, I am unwilling to vouch for the truth of that which I think I found in Western Australia on the night of July 17–18, 1935. There is reason to hope that my experience was wholly or partly an hallucination—for which, indeed, abundant causes existed. And yet, its realism was so hideous that I sometimes find hope impossible. If the thing did happen, then man must be prepared to accept notions of the cosmos, and of his own place in the seething vortex of time, whose merest mention is paralysing. He must, too, be placed on guard against a specific lurking peril which, though it will never engulf the whole race, may impose monstrous and unguessable horrors upon certain venturesome members of it. It is for this latter reason that I urge, with all the force of my being, a final abandonment of all attempts at unearthing those fragments of unknown, primordial masonry which my expedition set out to investigate.

Put `@import` 1st so it can be overridden

Don't use it unless you absolutely need to use it, as it can slow down your page loads[*]
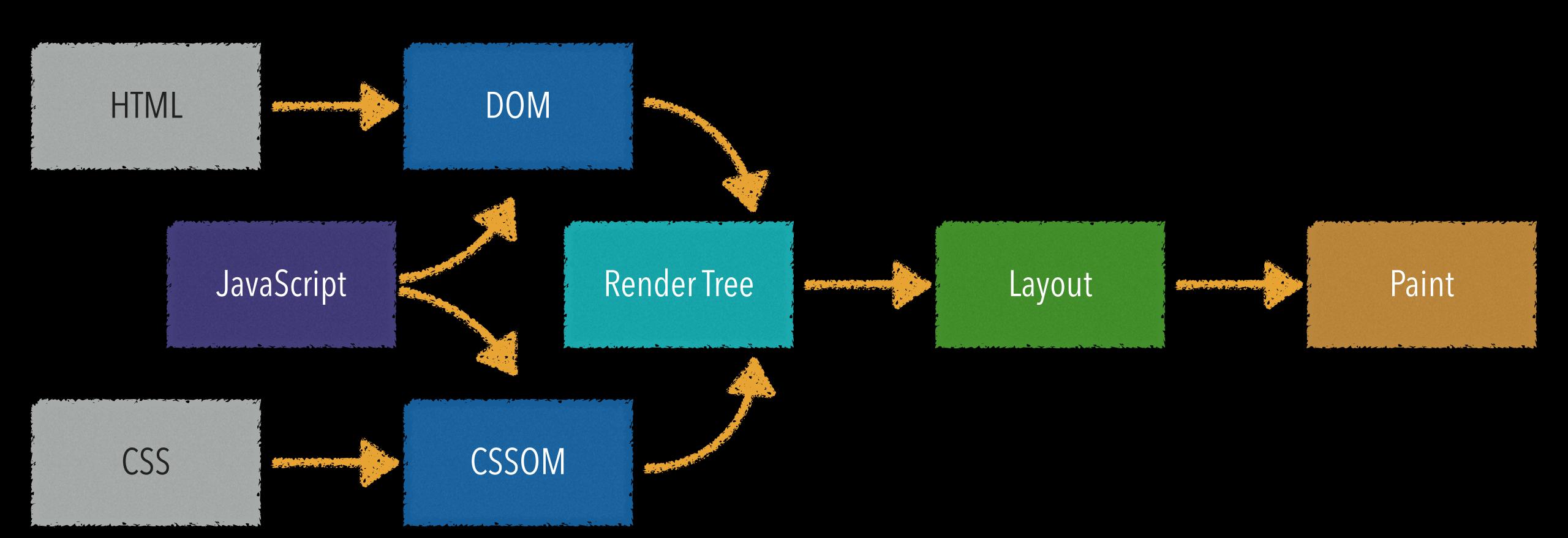
📝 SIDE NOTE

Delay	User Perception
0–100 ms	Instant
100–300 ms	Small perceptible delay
300–1,000 ms	Machine is working
1,000+ ms	Likely mental context switch
10,000+ ms	Task abandoned

The Browser Processing Pipeline

```
HTML ──────▶ DOM ──────╮
                        │
                        ▼
JavaScript ──────▶ Render Tree ──────▶ Layout ──────▶ Paint
          ──────▶              ▲
                               │
CSS ──────▶ CSSOM ─────────────╯
```
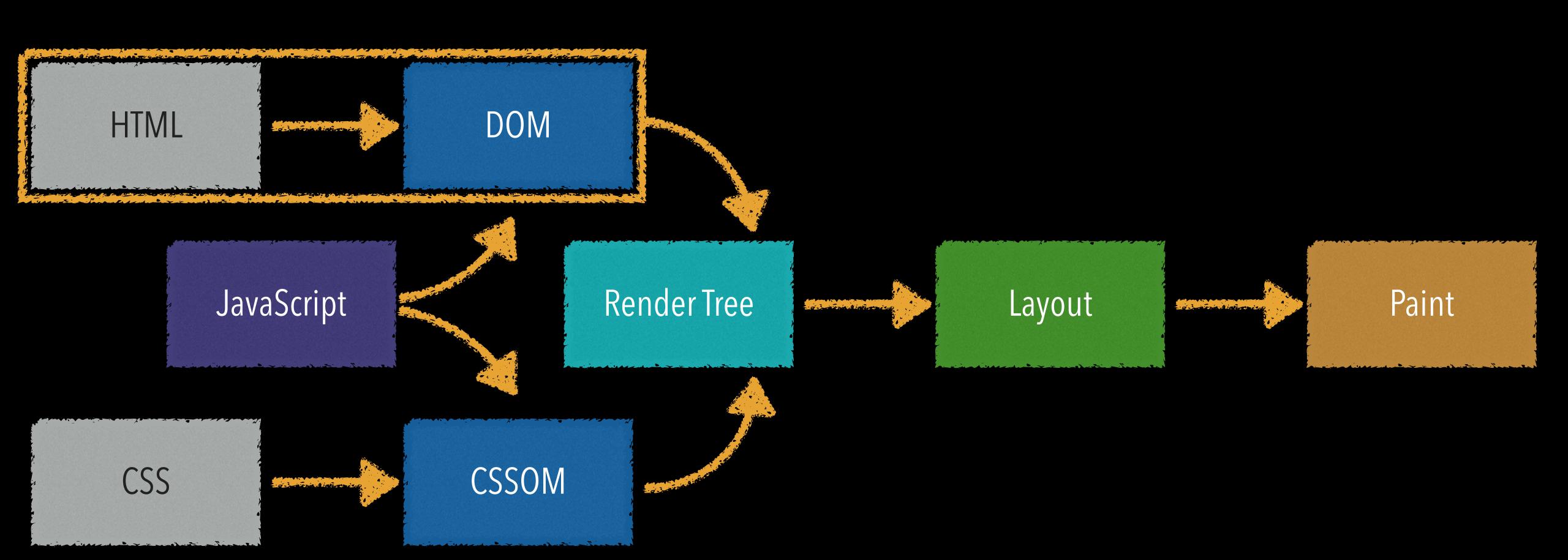
To understand CSS, you have to understand

» the DOM (*Document Object Model*)
» the CSSOM (*CSS Object Model*)
» the browser processing pipeline

The DOM

Before a webpage appears in a viewport, the rendering engine downloads the HTML & parses it to figure out how to display the webpage on screen

During this process, the rendering engine creates the DOM tree

"A Web page is a document. This document can be either displayed in the browser window, or as the HTML source. But it is the same document in both cases. The Document Object Model (DOM) provides another way to represent, store and manipulate that same document. *The DOM is a fully object-oriented representation of the web page, and it can be modified with a scripting language such as JavaScript.*" —Mozilla Developer Network

When is the DOM different than your HTML?

If you have mistakes in your HTML, the rendering engine "fixes" them when it generates the DOM

Article   Talk   Read   Edit   View history   Search Wikipedia 🔍

The Thing (1982 film)

From Wikipedia, the free encyclopedia

> 📖❓ This article **needs additional citations for verification**. Please help improve this article by adding citations to reliable sources. Unsourced material may be challenged and removed.
> *(May 2016)* *(Learn how and when to remove this template message)*

The Thing (also known as **John Carpenter's The Thing**) is a 1982 American science-fiction horror film directed by John Carpenter, written by Bill Lancaster, and starring Kurt Russell. The film's title refers to its primary antagonist: a parasitic extraterrestrial lifeform that assimilates other organisms and in turn imitates them. The Thing infiltrates an Antarctic research station, taking the appearance of the researchers that it absorbs, and paranoia develops within the group.

The film is based on John W. Campbell, Jr.'s novella *Who Goes There?*, which was more loosely adapted by Howard Hawks and Christian Nyby as the 1951 film *The Thing from Another World*.[3] Carpenter considers *The Thing* to be the first part of his *Apocalypse Trilogy*,[4] followed by *Prince of Darkness* and *In the Mouth of Madness*. Although the films are narratively unrelated, each features a potentially apocalyptic scenario; should "The Thing" ever reach civilization, it would be only a matter of time before it consumes all life on Earth. Carpenter acknowledged that the work of H.P. Lovecraft also inspired the film.[5][6]

On June 25, 1982, *The Thing* opened in 840 theaters, and it was the #8 movie of the

The Thing

The ultimate in alien terror.

THING

Sidebar (left navigation)

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction

Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools

What links here
Related changes
Upload file
Special pages
Permanent link

Source Code

```html
1 <!DOCTYPE html>
2 <html class="client-nojs" lang="en" dir="ltr">
3 <head>
4 <meta charset="UTF-8"/>
5 <title>The Thing (1982 film) – Wikipedia</title>
6 <script>document.documentElement.className = document.documentElement.className.replace( /(^|\s)client-nojs(\s|$)/, "$1client-js$2" );</script>
7 <script>(window.RLQ=window.RLQ||[]).push(function()
{mw.config.set({"wgCanonicalNamespace":"","wgCanonicalSpecialPageName":false,"wgNamespaceNumber":0,"wgPageName":"The_Thing_(1982_film)","wgTitle":"The Thing (1982
film)","wgCurRevisionId":806123497,"wgRevisionId":806123497,"wgArticleId":30809,"wgIsArticle":true,"wgIsRedirect":false,"wgAction":"view","wgUserName":null,"wgUserGroups":["*"],"wgCategories":["CS1 maint: Unfit url","Webarchive template wayback
links","Pages with citations lacking titles","Pages with citations having bare URLs","Articles needing additional references
from May 2016","All articles needing additional references","Articles needing additional references from February
2017","Articles needing additional references from April 2017","All articles with unsourced statements","Articles with
unsourced statements from August 2016","Articles with hAudio microformats","Wikipedia articles with VIAF
identifiers","Wikipedia articles with GND identifiers","Wikipedia articles with BNF identifiers","1982 films","English-
language films","1982 horror films","1980s science fiction films","1980s science fiction horror films","Films based on science
fiction short stories","American films","American science fiction horror films","Norwegian-language films","Films directed by
John Carpenter","Action horror films","Films based on science fiction novels","Films set in Antarctica","Films set in
1982","Films shot in Alaska","Films shot in British Columbia","Monster movies","Films using stop-motion animation","Universal
Pictures films","Films about shapeshifting","Fictional parasites and parasitoids","Film scores by Ennio Morricone","Splatter
films","Body horror films","Films about extraterrestrial life","Films adapted into novels","Films adapted into comics","Films
adapted into video
games"],"wgBreakFrames":false,"wgPageContentLanguage":"en","wgPageContentModel":"wikitext","wgSeparatorTransformTable":
["",""],"wgDigitTransformTable":["",""],"wgDefaultDateFormat":"dmy","wgMonthNames":
["","January","February","March","April","May","June","July","August","September","October","November","December"],"wgMonthNamesShort":
["","Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","Nov","Dec"],"wgRelevantPageName":"The_Thing_(1982_film)","wgRelevantArticleId":30809,"wgRequestId":"WekWTQpAICgAAC6@iyoAAABT","wgIsProbablyEditable":true,"wgRelevantPageIsProbablyEditable":true,"wgRestrictionEdit":[],"wgRestrictionMove":[],"wgFlaggedRevsParams":{"tags":
{}},"wgStableRevisionId":null,"wgWikiEditorEnabledModules":
{"toolbar":true,"preview":false,"publish":false},"wgBetaFeaturesFeatures":
```

```
<!DOCTYPE html>
<html class="client-js ve-not-available" lang="en" dir="ltr">
  ▶ <head>…</head>
  ▶ <body class="mediawiki ltr sitedir-ltr mw-hide-empty-elt ns-0 ns-subject page-The_Thing_1982_film rootpage-The_Thing_1982_film
  vector-experimental-print-styles vector-nav-directionality skin-vector action-view">…</body>
</html> = $0
```

DOM as rendered by browser Inspector

Source Code

No `<tbody>`

```
41          <div id="bodycontent" class="mw-body-content">
42              <div id="siteSub" class="noprint">From Wikipedia, the free encyclopedia</div>
43          <div id="contentSub"></div>
44              <div id="jump-to-nav" class="mw-jump">
45      Jump to:          <a href="#mw-head">navigation</a>,          <a href="#p-search">search</a>
46          </div>
47          <div id="mw-content-text" lang="en" dir="ltr" class="mw-content-ltr"><div class="mw-parser-output"><table
   class="plainlinks metadata ambox ambox-content ambox-Refimprove" role="presentation">
48  <tr>
49  <td class="mbox-image">
50  <div style="width:52px"><a href="/wiki/File:Question_book-new.svg" class="image"><img alt=""
   src="//upload.wikimedia.org/wikipedia/en/thumb/9/99/Question_book-new.svg/50px-Question_book-new.svg.png" width="50"
   height="39" srcset="//upload.wikimedia.org/wikipedia/en/thumb/9/99/Question_book-new.svg/75px-Question_book-new.svg.png 1.5x,
   //upload.wikimedia.org/wikipedia/en/thumb/9/99/Question_book-new.svg/100px-Question_book-new.svg.png 2x" data-file-width="512"
   data-file-height="399" /></a></div>
51  </td>
52  <td class="mbox-text">
53  <div class="mbox-text-span">This article <b>needs additional citations for <a href="/wiki/Wikipedia:Verifiability"
   title="Wikipedia:Verifiability">verification</a></b>. <span class="hide-when-compact">Please help <a class="external text"
   href="//en.wikipedia.org/w/index.php?title=The_Thing_(1982_film)&amp;action=edit">improve this article</a> by <a
   href="/wiki/Help:Introduction_to_referencing_with_Wiki_Markup/1" title="Help:Introduction to referencing with Wiki
   Markup/1">adding citations to reliable sources</a>. Unsourced material may be challenged and removed.</span> <small><i>(May
   2016)</i></small> <small class="hide-when-compact"><i>(<a href="/wiki/Help:Maintenance_template_removal"
   title="Help:Maintenance template removal">Learn how and when to remove this template message</a>)</i></small></div>
54  </td>
55  </tr>
56  </table>
57  <table class="infobox vevent" style="width:22em;font-size:90%;">
58  <tr>
59  <th colspan="2" class="summary" style="text-align:center;font-size:125%;font-weight:bold;font-size:110%;font-
   style:italic;">The Thing</th>
60  </tr>
61  <tr>
```
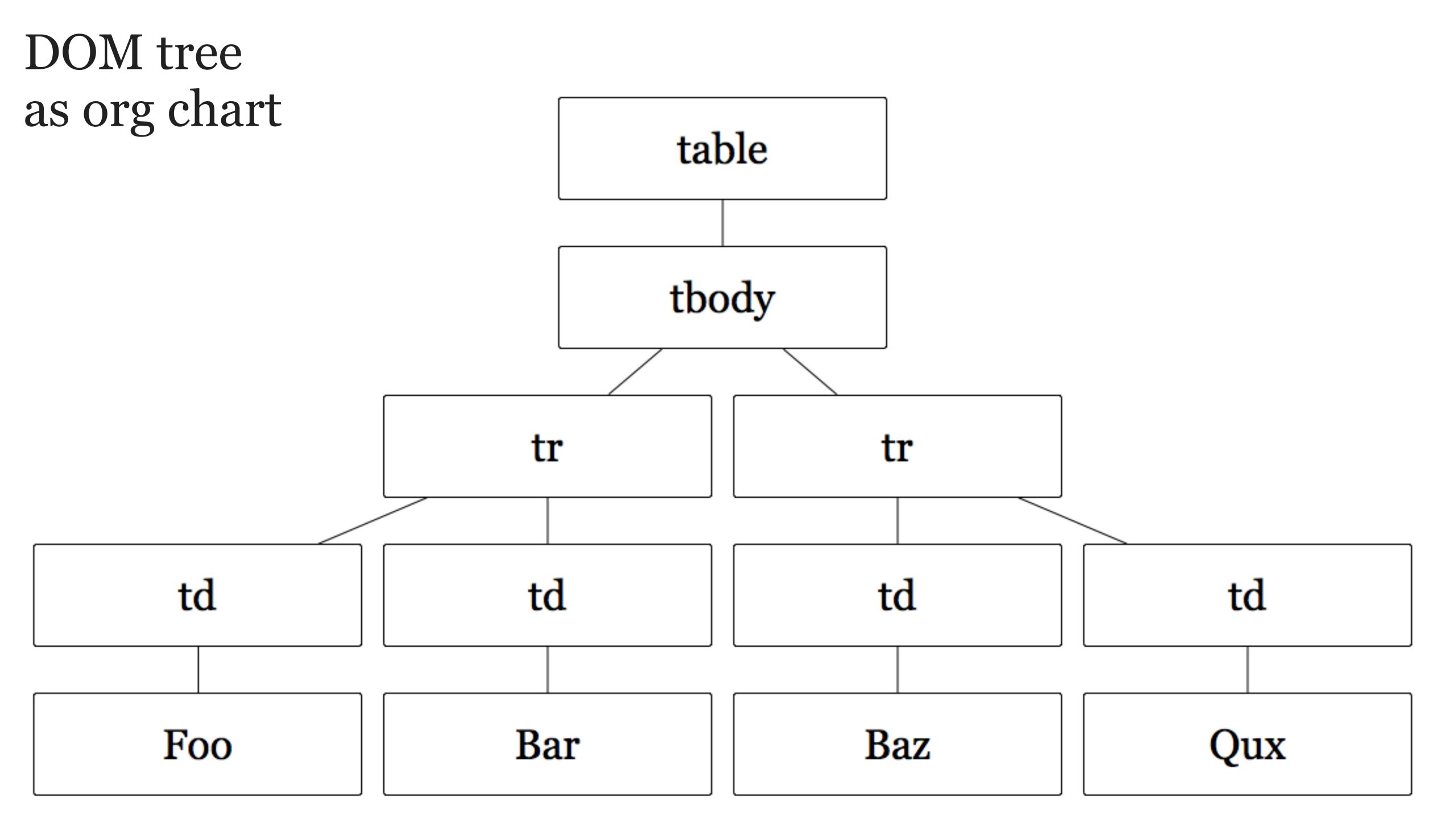
```
▼<body class="mediawiki ltr sitedir-ltr mw-hide-empty-elt ns-0 ns-subject page-The_Thing_1982_film rootpage-The_Thing_1982_film
vector-experimental-print-styles vector-nav-directionality skin-vector action-view">
    <div id="mw-page-base" class="noprint"></div>
    <div id="mw-head-base" class="noprint"></div>
  ▼<div id="content" class="mw-body" role="main">
      <a id="top"></a>
    ▶<div class="mw-body-content" id="siteNotice">…</div>
      <div class="mw-indicators mw-body-content"> </div>
    ▶<h1 id="firstHeading" class="firstHeading" lang="en">…</h1>
    ▼<div id="bodyContent" class="mw-body-content">
        <div id="siteSub" class="noprint">From Wikipedia, the free encyclopedia</div>
        <div id="contentSub"></div>
      ▶<div id="jump-to-nav" class="mw-jump">…</div>
      ▼<div id="mw-content-text" lang="en" dir="ltr" class="mw-content-ltr">
        ▼<div class="mw-parser-output">
          ▼<table class="plainlinks metadata ambox ambox-content ambox-Refimprove" role="presentation">
            ▼<tbody>
              ▼<tr>
                ▼<td class="mbox-image">
                  ▼<div style="width:52px">
                    ▼<a href="/wiki/File:Question_book-new.svg" class="image">
                        <img alt src="//upload.wikimedia.org/wikipedia/en/thumb/9/99/Question_book-
                        new.svg.png" width="50" height="39" srcset="//upload.wikimedia.org/wikipedia/en/thumb/9/99/
                        Question_book-new.svg/75px-Question_book-new.svg.png 1.5x, //upload.wikimedia.org/wikipedia/en/thumb/
                        9/99/Question_book-new.svg/100px-Question_book-new.svg.png 2x" data-file-width="512" data-file-height=
                        "399">
                    </a>
                  </div>
                </td>
              ▶<td class="mbox-text">…</td>
              </tr>
            </tbody>
          </table>
```

`<tbody>` added

Rendered DOM

```
<table>
  <tr>
    <td>Foo</td>
    <td>Bar</td>
  </tr>
  <tr>
    <td>Baz</td>
    <td>Qux</td>
  </tr>
</table>
```

Source code

```
<table>
  <tbody>
    <tr>
      <td>Foo</td>
      <td>Bar</td>
    </tr>
    <tr>
      <td>Baz</td>
      <td>Qux</td>
    </tr>
  </tbody>
</table>
```

DOM tree

DOM tree
as org chart

```
                        ┌─────────────┐
                        │    table    │
                        └──────┬──────┘
                        ┌──────┴──────┐
                        │    tbody    │
                        └──────┬──────┘
                 ┌─────────────┴─────────────┐
            ┌────┴────┐                 ┌─────┴────┐
            │   tr    │                 │    tr    │
            └──┬───┬──┘                 └──┬────┬──┘
        ┌──────┘   └────┐           ┌──────┘    └──────┐
   ┌────┴───┐      ┌────┴───┐   ┌───┴────┐         ┌───┴────┐
   │   td   │      │   td   │   │   td   │         │   td   │
   └────┬───┘      └────┬───┘   └────┬───┘         └────┬───┘
   ┌────┴───┐      ┌────┴───┐   ┌────┴───┐         ┌────┴───┐
   │  Foo   │      │  Bar   │   │  Baz   │         │  Qux   │
   └────────┘      └────────┘   └────────┘         └────────┘
```

DOM tree as nested boxes

table
  tbody
    tr
      td
        Foo
      td
        Bar
    tr
      td
        Baz
      td
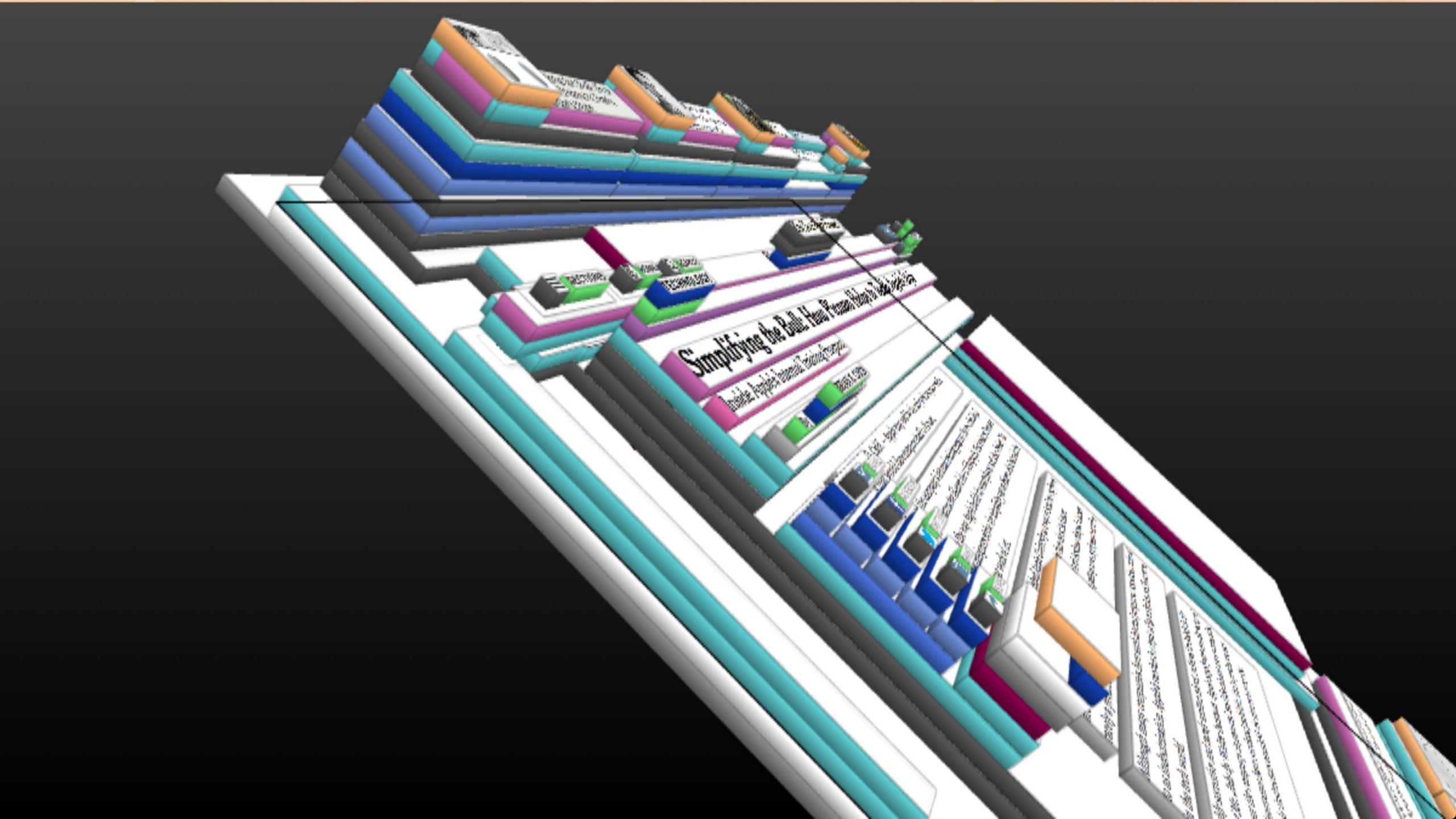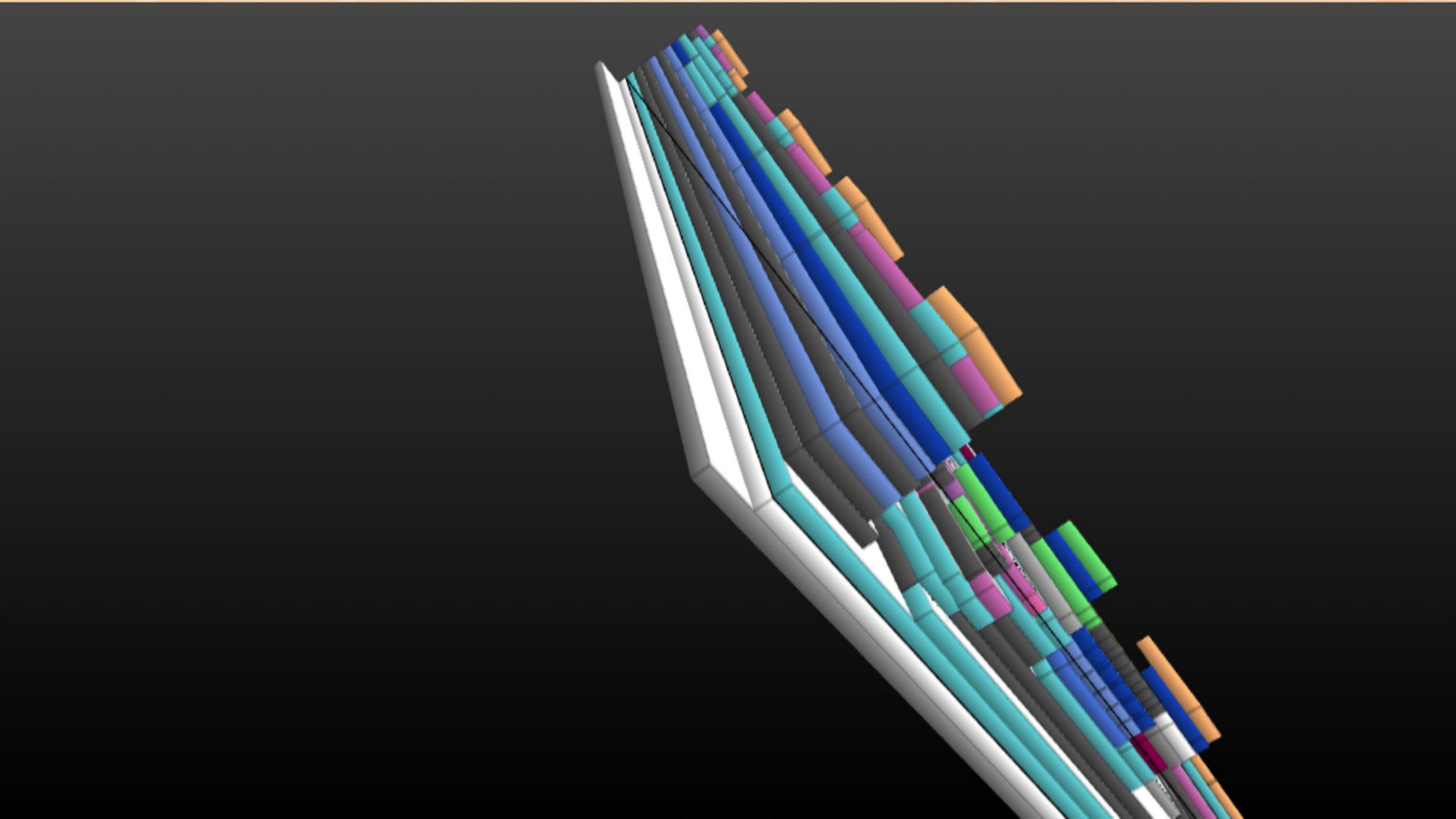   **Qux**

Firefox up to version 47 had a cool feature called *3D View* that let you "rotate and re-orient the 3D presentation of the DOM hierarchy of your page to see it from different angles"

Simplifying the Bull: How Picasso Helps to Teach Apple's Style

CUPERTINO, Calif. — Apple may well be the only tech company on the planet that would dare compare itself to Picasso.

In a class at the company's internal training program, the so-called Apple University, the instructor likened the 11 lithographs that make up Picasso's "The Bull" to the way Apple builds its smartphones and other devices. The idea: Apple designers strive for simplicity just as Picasso eliminated details to create a great work of art.

Steven P. Jobs established Apple University as a way to inculcate employees into Apple's business culture and educate them about its history, particularly as the company grew and the tech business changed. Courses are not required, only recommended, but getting new employees to enroll is rarely a problem.

Although many companies have such internal programs, sometimes referred to as indoctrination, Apple's version is a topic of speculation and fascination in the tech world.
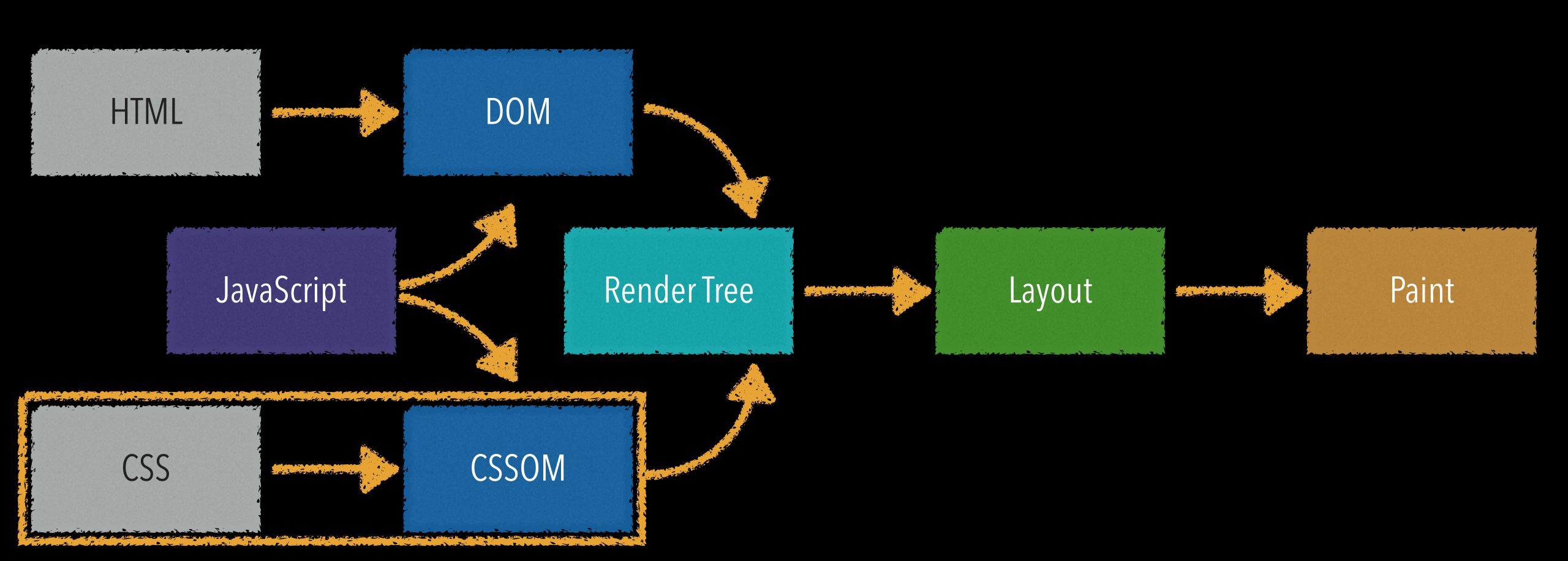
It is highly secretive and rarely written about, referred to briefly in the
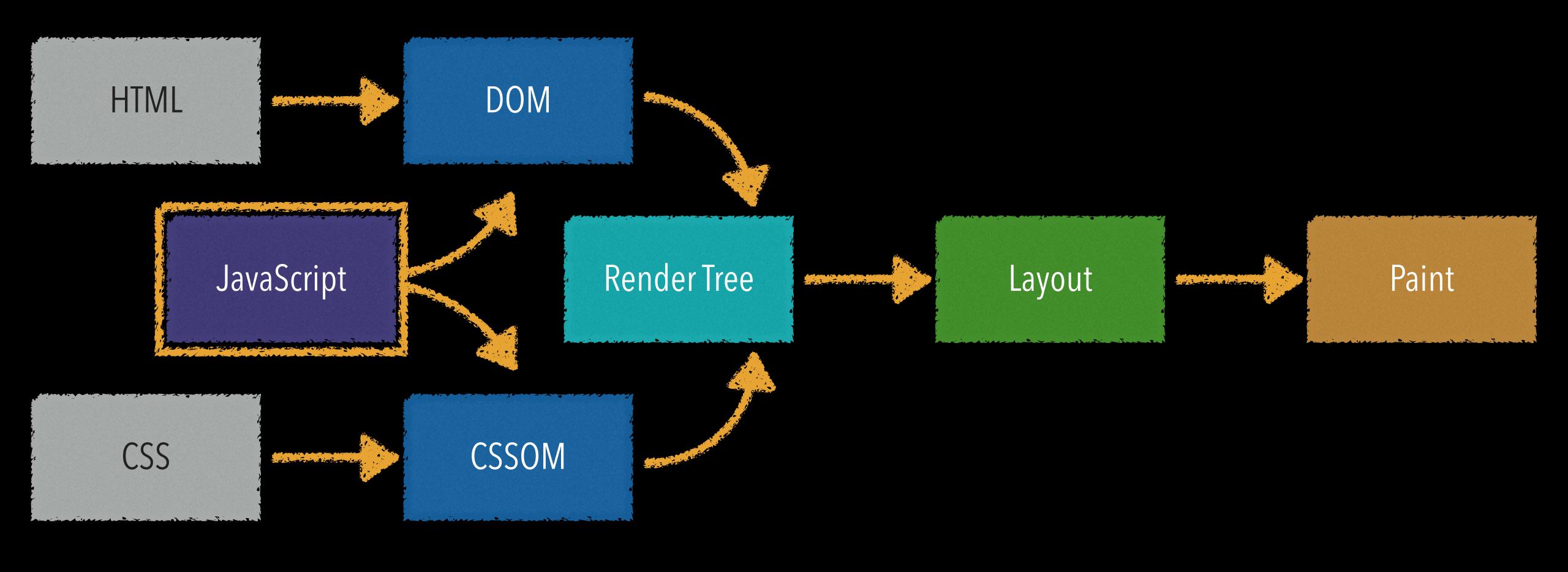
Simplifying the Build. How Power Works Shapes

LOG IN

COMMENTS

Amazon Te...
Muppets Off the...

Simplifying the Bull: How Picasso Helps to Teach Apple's Style

The CSSOM

The CSSOM is built by the rendering engine using specified stylesheet rules from:

» built-in rules that come with the *browser*
» rules added by *user*
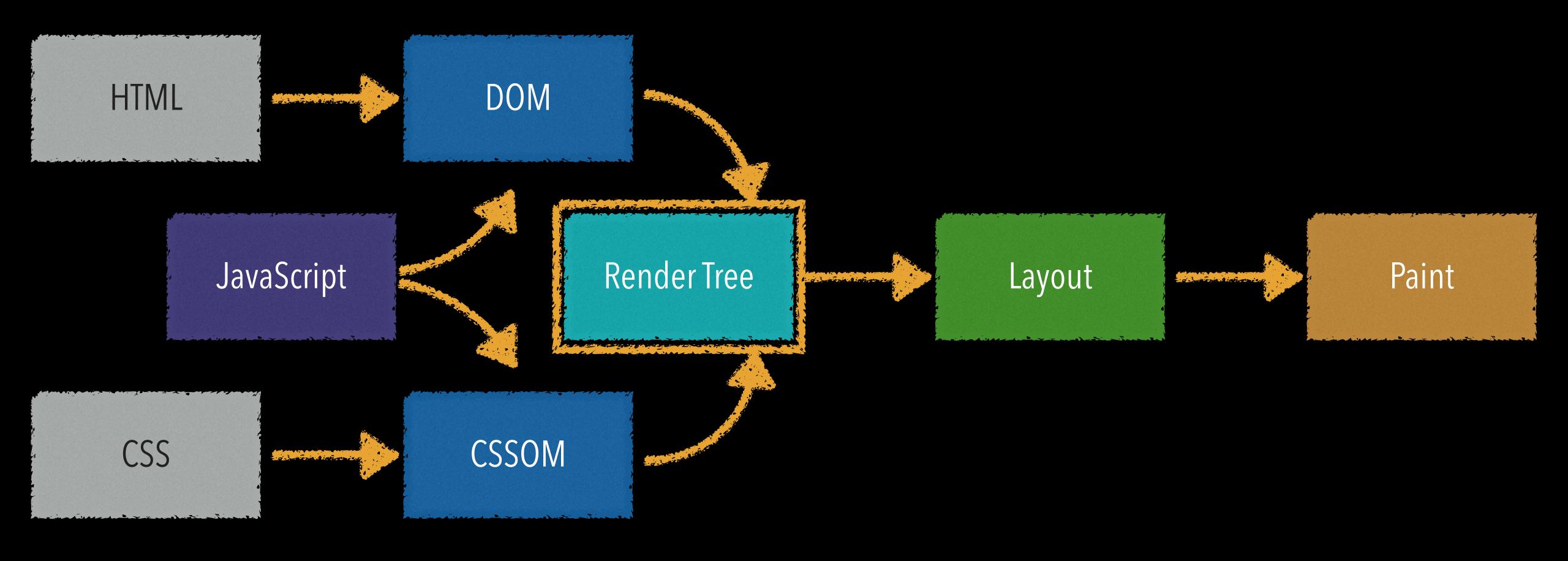» rules created by CSS *authors* (developers)
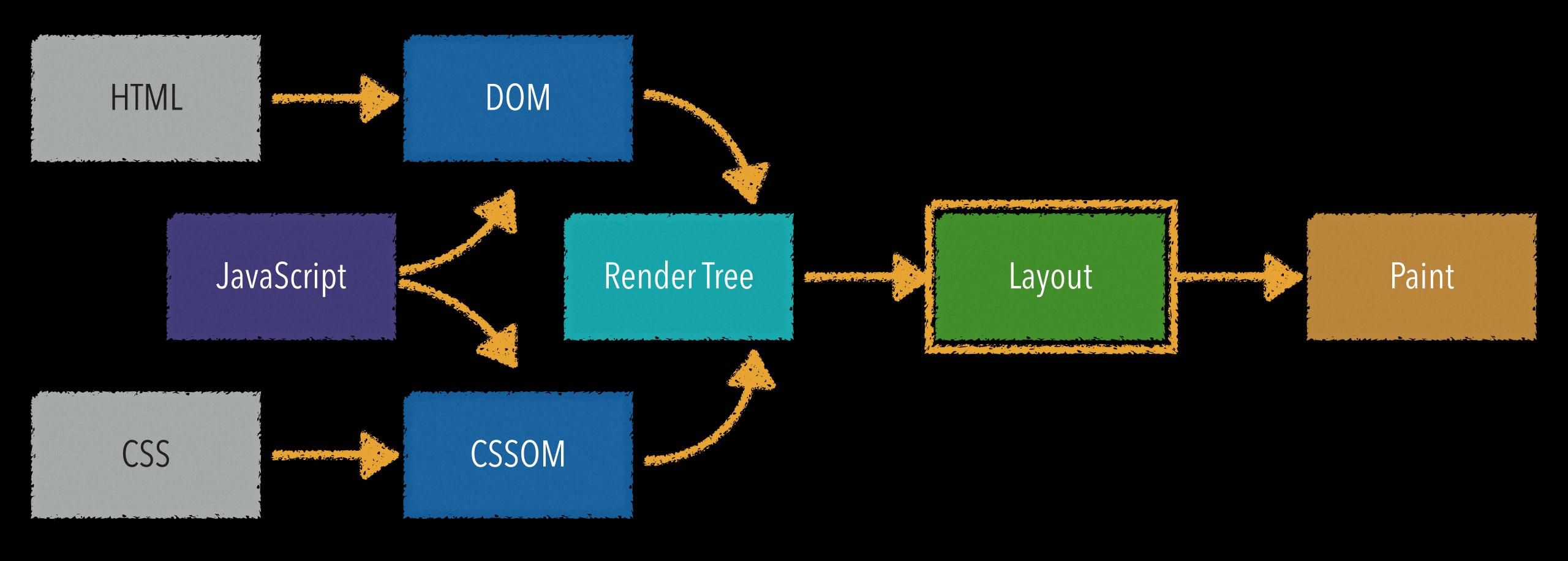
Bringing It All Together

JavaScript can change the DOM & the CSSOM

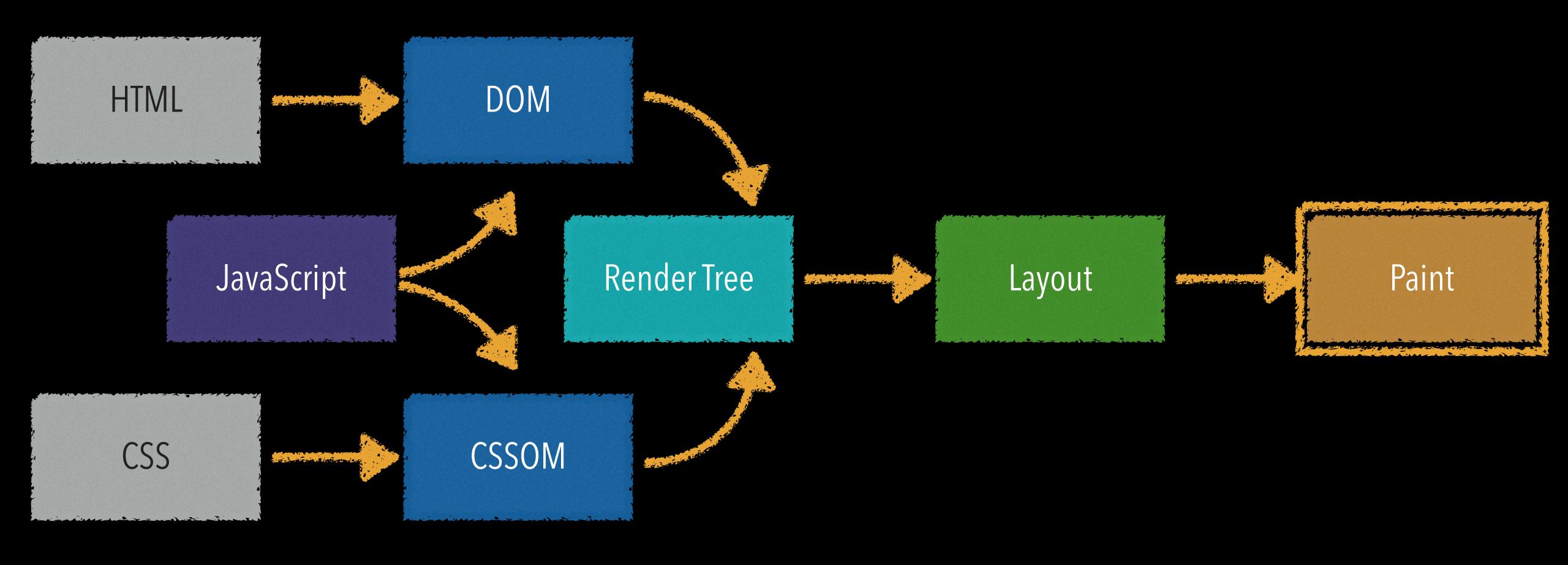As of July 2020, there are 69 Web APIs

Ambient Light Events • Background Tasks • Battery • Beacon • Bluetooth • Broadcast Channel • CSS Counter Styles • CSS Font Loading • CSSOM • Canvas • Channel Messaging • Console • Credential Management • DOM • Encoding • Encrypted Media Extensions • Fetch • File System • Frame Timing • Fullscreen • Gamepad • Geolocation • HTML Drag and Drop • High Resolution Time • History • Image Capture • IndexedDB • Intersection Observer • Long Tasks • Media Capabilities • Media Capture and Streams • Media Session • Media Source Extensions • MediaStream Recording • Navigation Timing • Network Information • Page Visibility • Payment Request • Performance • Performance Timeline • Permissions • Pointer Events • Pointer Lock • Proximity Events • Push • Resize Observer • Resource Timing • Server Sent Events • Service Workers • Storage • Storage Access • Streams • Touch Events • URL • Vibration • Visual Viewport • Web Animations • Web Audio • Web Authentication • Web Crypto • Web Notifications • Web Storage • Web Workers • WebGL • WebRTC • WebVR • WebVTT • WebXR Device • Websockets
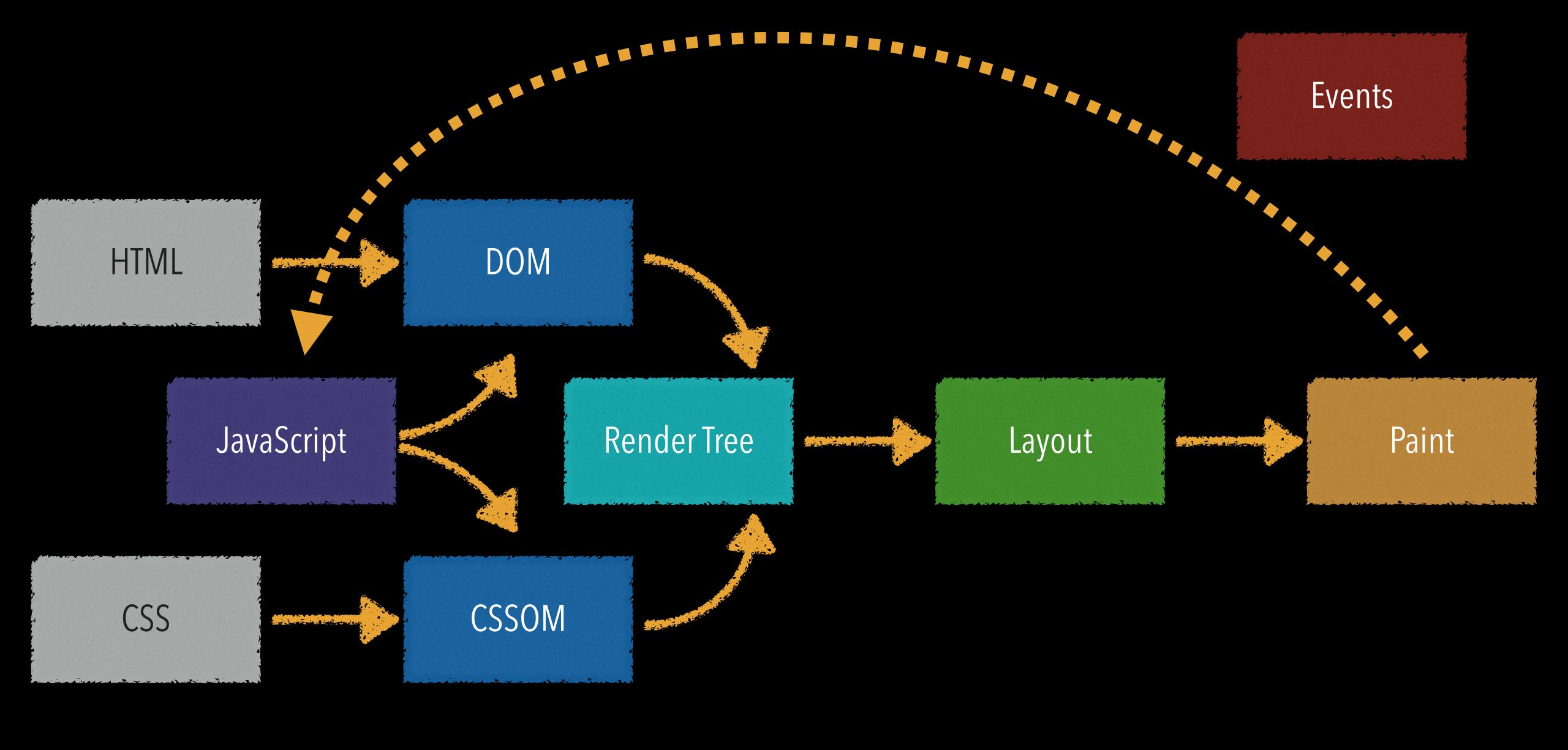
The rendering engine matches HTML element objects
with CSS rule objects to generate the Render Tree

The rendering engine uses the Render Tree to layout boxes in the viewport

The rendering engine displays (*paints*) all the content of the page in the viewport, from back to front

JavaScript events can further change the DOM & the CSSOM

The big takeaway: when you're working with CSS, you are manipulating objects that will be rendered as boxes

Basic Selectors

A CSS *selector* declares which DOM objects should have particular styles applied to them

The browser's rendering engine…
» parses the CSS & HTML
» matches selectors to the appropriate DOM objects
» applies the CSS style to the rendered DOM objects

HTML	CSS
`<h1 align="center">`	`h1 {text-align: center;}`
Entire thing is a *tag*	Entire thing is a *ruleset*
`h1` is an *element*	`h1` is a *selector*
`align` is an *attribute*	`text-align` is a *property*
`center` is a *value*	`center` is a *value*
`align="center"` is an *attribute-value pair*	`text-align: center` is a *declaration*
	Everything inside `{` & `}` is a *declaration block*

CSS 1: 10 different selectors (including selectors, *combinators*, *pseudo-classes*, & *pseudo-elements*)

CSS 2: 13

CSS 3: 21

CSS 4: 26 (so far!)

70 in total

1. Simple selectors

» Universal
» Type
» Class
» ID
» Pseudo-classes
» Pseudo-elements
» Attribute

2. Compound selectors

3. Complex selectors with combinators

» Descendant
» Child
» Adjacent sibling
» General sibling

4. Selector list

Simple Selectors

A *simple selector* describes a *single condition* on an element

If the condition is true, the element is selected

`*` (universal): Is this an element?
`element`: Is this the specific type of element?
`.class`: Does this element have this class on it?
`#ID`: Does this element have this ID on it?

Universal

*

Selects *every HTML element*

(Though later CSS rules can override these declarations)

HTML

```html
    sisters had come to look very queer,
    especially the elder ones; and it was
    said that their health was failing.
7  </p>
8  <hr>
9  <p>
10   After reading, also check out:
11 </p>
12 <ul>
13   <li>Dagon</li>
14   <li>The Terrible Old Man</li>
15   <li>The Colour Out of Space</li>
16 </ul>
```

CSS

```css
1  * {
2    margin: 0;
3    padding: 0;
4  }
5
6  /* Uninteresting stuff below here */
7
8  html {
9    font-size: 24px;
10   line-height: 1.4;
11   font-family: "Iowan Old Style",
   Georgia, serif;
```

JS

From "The Shadow Over Innsmouth" by H. P. Lovecraft

Of course the town's real business was the refinery, whose commercial office was on the square only a few doors east of where we stood. Old Man Marsh was never seen, but sometimes went to the works in a closed, curtained car.

There were all sorts of rumours about how Marsh had come to look. He had once been a great dandy, and people said he still wore the frock-coated finery of the Edwardian age, curiously adapted to certain deformities. His sons had formerly conducted the office in the square, but latterly they had been keeping out of sight a good deal and leaving the brunt of affairs to the younger generation. The sons and their sisters had come to look very queer, especially the elder ones; and it was said that their health was failing.

After reading, also check out:

Dagon

The Terrible Old Man

The Colour Out of Space

Type

`element`

Selects *every matching HTML element* (e.g., `<p>`, `<ul>`, or `<h3>`)

AKA *Element Selectors*

Used when you want to affect every instance of an element

HTML

```html
<h3>H. P. Lovecraft's "The Dunwich
Horror"</h3>
<p>
  About 1923, when Wilbur was a boy of
  ten whose mind, voice, stature, and
  bearded face gave all the impressions
  of maturity, a second great siege of
  carpentry went on at the old house. It
  was all inside the sealed upper part,
  and from bits of discarded lumber
  people concluded that the youth and
  his grandfather had knocked out all
  the partitions and even removed the
  attic floor, leaving only one vast
```

CSS

```css
p {
  color: red;
  font-weight: bold;
}

/* Uninteresting stuff below here */

html {
  font-size: 24px;
  font-family: "Iowan Old Style",
  Georgia, serif;
}
```

JS

H. P. Lovecraft's "The Dunwich Horror"
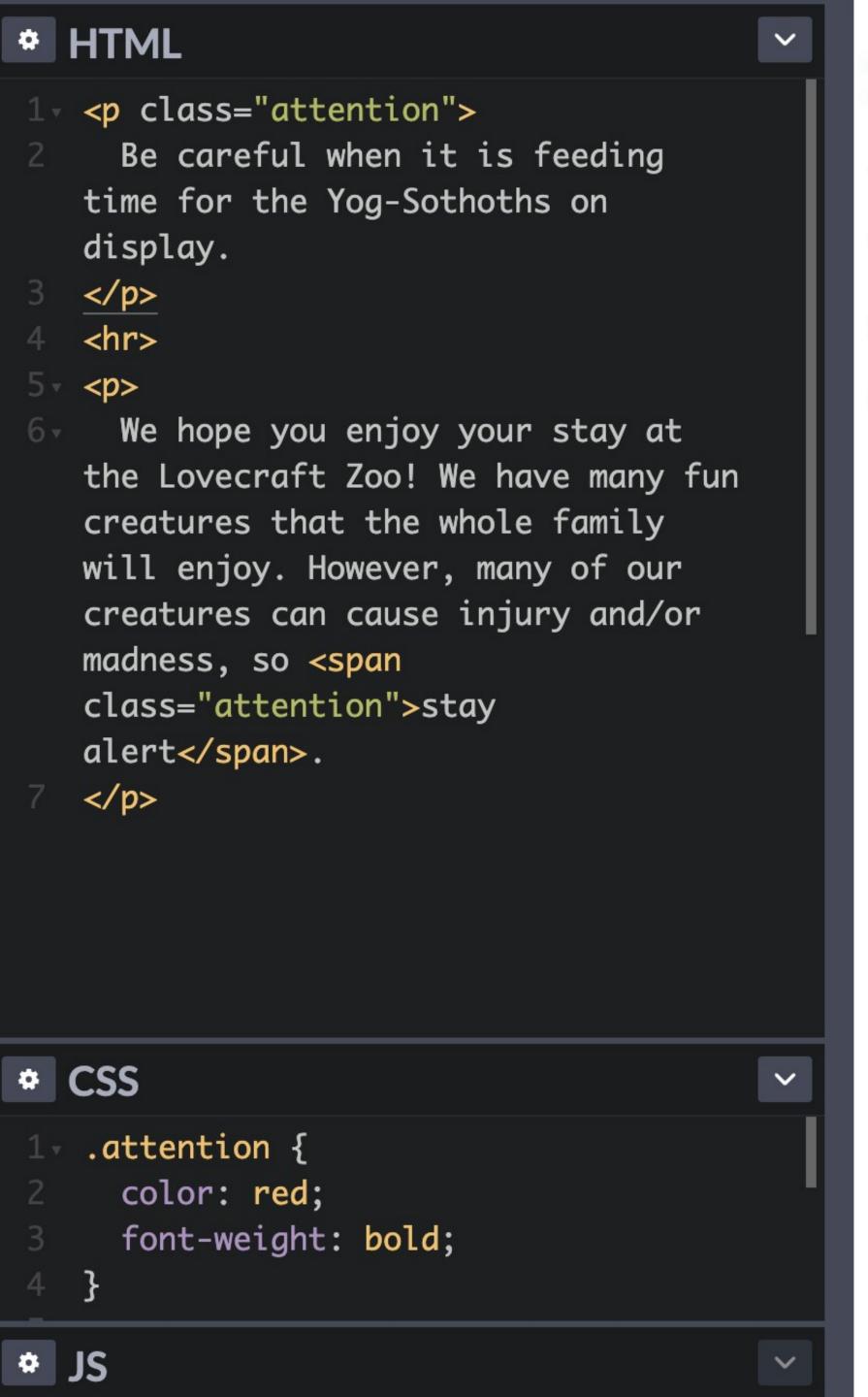
About 1923, when Wilbur was a boy of ten whose mind, voice, stature, and bearded face gave all the impressions of maturity, a second great siege of carpentry went on at the old house. It was all inside the sealed upper part, and from bits of discarded lumber people concluded that the youth and his grandfather had knocked out all the partitions and even removed the attic floor, leaving only one vast open void between the ground story and the peaked roof. They had torn down the great central chimney, too, and fitted the rusty range with a flimsy outside tin stovepipe.

In the spring after this event Old Whateley noticed the growing number of whippoorwills that would come out of Cold Spring Glen to chirp under his window at night. He seemed to regard the circumstance as one of great significance, and told the loungers at Osborn's that he thought his time had almost come.

Class

`.class`

Selects any *element to which the class has been applied*, as many times per page as needed

HTML

```html
<p class="attention">
    Be careful when it is feeding
time for the Yog-Sothoths on
display.
</p>
<hr>
<p>
    We hope you enjoy your stay at
the Lovecraft Zoo! We have many fun
creatures that the whole family
will enjoy. However, many of our
creatures can cause injury and/or
madness, so <span
class="attention">stay
alert</span>.
</p>
```

CSS

```css
.attention {
  color: red;
  font-weight: bold;
}
```

JS

Be careful when it is feeding time for the Yog-Sothoths on display.

We hope you enjoy your stay at the Lovecraft Zoo! We have many fun creatures that the whole family will enjoy. However, many of our creatures can cause injury and/or madness, so **stay alert**.

Dot in front of the class name in CSS, but no dot in HTML

Dot labels the class in CSS

The `class` attribute labels the class in HTML

```
HTML                                              ⌄

1▾ <p class="attention">
2      Be careful when it is feeding
   time for the Yog-Sothoths on
   display.
3    </p>
4    <hr>
5▾ <p>
6▾    We hope you enjoy your stay at
   the Lovecraft Zoo! We have many fun
   creatures that the whole family
   will enjoy. However, many of our
   creatures can cause injury and/or
   madness, so <span
   class="attention">stay
   alert</span>.
7    </p>
```

Be careful when it is feeding time for the Yog-Sothoths on display.

We hope you enjoy your stay at the Lovecraft Zoo! We have many fun creatures that the whole family will enjoy. However, many of our creatures can cause injury and/or madness, so **stay alert**.

```
CSS                                               ⌄

1▾ .attention {
2    color: red;
3    font-weight: bold;
4 }
```

The paragraph with `class` applied to it is different from the paragraph that does not have that `class`

```
JS                                                ⌄
```

Be careful when it is feeding time for the Yog-Sothoths on display.

We hope you enjoy your stay at the Lovecraft Zoo! We have many fun creatures that the whole family will enjoy. However, many of our creatures can cause injury and/or madness, so **stay alert**.

A class can be applied as many times per page as you wish

You can use multiple (2, 3, 4, whatever) classes on an element when needed

```
HTML                                        ⌄

1 ⌄ <figure>
2     <img src="https://s3-us-
    west-2.amazonaws.com/s.cdpn.io
    /122116/Libby.nov98.jpg"
    alt="Libby">
3 ⌄   <figcaption class="center
    image-caption">
4       Libby was a cute puppy
5     </figcaption>
6   </figure>
```


Libby was a cute puppy

```
CSS                                         ⌄

1 ⌄ .center {
2     text-align: center;
3   }
4
5 ⌄ .image-caption {
6     font-size: .9rem;
7     font-style: italic;
8   }
9
```

```
JS                                          ⌄
```

✏️ SIDE NOTE

A student once did this:

```
<figcaption class="center" class="image-caption"> 🤨
```

This will *not* work! No duplicate attributes!

You *must* use:

```
<figcaption class="center image-caption">
```

📝 SIDE NOTE

How do you match an element that has a particular combination of classes on it?

`<p class="center note">`

See *Compound Selectors* later in this slide deck

There is no default list of class names

You come up with the class names your project uses (or you use those provided by a framework like Bootstrap)

Don't use spaces

Rules for class values in HTML5

In fact, you can use emoji for class names

```
.📰 {
  background-color: hsl(0,0%,76%);
}


.🤮 {
  font-family: "Comic Sans", cursive;
}
```

If you use emoji, you must put `@charset "utf-8";` at the top of your style sheet

The big rule for class names: *describe function, not appearance*

Not "what does it look like?" but instead "what is it doing?"

Bad class names

```
.author name
.big-red
.small
```

Good class names

```
.author-name
.caption
.alert
.footnote
.center
```

Multiple words in a class name?

```
.main-content-nav   ← What most developers use
.main_content_nav
.maincontentnav
.mainContentNav
.MainContentNav
```

Just be consistent!

ID

#id

Selects any *element to which the ID has been applied*, but each ID can only be used once per page

You can use multiple IDs on a page, but each ID must be unique

`#id` & `.class` share a lot of similarities

» Names are made up by developers, not specs
» Names shouldn't have spaces
» Names should denote purpose, not appearance
» Names containing multiple words should be combined with a `-` or something else consistently

HTML

```html
1  <article>
2    <h3>"The Shadow Over Innsmouth"
   by H. P. Lovecraft</h3>
3    <p>↔</p>
6    <aside id="pull-quote">
7      There were all sorts of
   rumours about how Marsh had come
   to look.
8    </aside>
9    <p>↔</p>
12  </article>
13
14  <aside id="sidebar">
15    <h3>August Derleth</h3>
16    <p>↔</p>
19  </aside>
20
21
22
23
24
25
26
27
28
```

CSS

JS

"The Shadow Over Innsmouth" by H. P. Lovecraft

Of course the town's real business was the refinery, whose commercial office was on the square only a few doors east of where we stood. Old Man Marsh was never seen, but sometimes went to the works in a closed, curtained car.

There were all sorts of rumours about how Marsh had come to look.

There were all sorts of rumours about how Marsh had come to look. He had once been a great dandy, and people said he still wore the frock-coated finery of the Edwardian age, curiously adapted to certain deformities. His sons had formerly conducted the office in the square, but latterly they had been keeping out of sight a good deal and leaving the brunt of affairs to the younger generation. The sons and their sisters had come to look very queer, especially the elder ones; and it was said that their health was failing.

August Derleth

Best remembered as the first publisher of the writings of H. P. Lovecraft, and for his own contributions to the Cthulhu Mythos genre of horror.

```html
 1  <article>
 2      <h3>"The Shadow Over Innsmouth"
    by H. P. Lovecraft</h3>
 3      <p>↔</p>
 6      <aside id="pull-quote">
 7          There were all sorts of
    rumours about how Marsh had come
    to look.
 8      </aside>
 9      <p>↔</p>
12  </article>
13
14  <aside id="sidebar">
```

```css
11  #sidebar {
12      width: 11rem;
13      margin: 0 1em;
14      padding: 1em;
15      box-sizing: border-box;
16  }
17
18  #pull-quote {
19      float: left;
20      width: 12em;
21      margin: .4em .75em 0 0;
```

"The Shadow Over Innsmouth" by H. P. Lovecraft

Of course the town's real business was the refinery, whose commercial office was on the square only a few doors east of where we stood. Old Man Marsh was never seen, but sometimes went to the works in a closed, curtained car.

There were all sorts of rumours about how Marsh had come to look.

There were all sorts of rumours about how Marsh had come to look. He had once been a great dandy, and people said he still wore the frock-coated finery of the Edwardian age, curiously adapted to certain deformities. His sons had formerly conducted the office in the square, but latterly they had been keeping out of sight a good deal and leaving the brunt of affairs to the younger generation. The s... ...lly ...n...ad come to look very queer, e... ...it was said that their healt...

August Derleth

Best remembered as the first publisher of the writings of H. P. Lovecraft, and for his own contributions to the Cthulhu Mythos genre of horror.

A specific ID can be applied one time per page

	Class	ID
CSS	`.foo`	`#bar`
HTML	`class="foo"`	`id="bar"`
Unique on page	No `class="foo"` 10x	Yes `id="bar"` once
Values per element	Multiple `class="foo bar"`	One `id="baz"`
Specificity	10×	100×
URL addressable	No	Yes

Bad ID names

`#top nav`
`#tiny-little-fonts`

Good ID names

`#sidebar` (use `<aside>`)
`#utility-nav` (use `<nav>`)
`#site-footer` (use
`<footer>`)
`#logo`
`#legalese`

Use classes instead of IDs (in fact, try to avoid IDs as much as possible when it comes to CSS)*

Classes can be reused, while IDs cannot

IDs can make the *cascade* (more about that soon!) very complicated

Many (most?) frameworks (like Bootstrap) stick to classes entirely

* OK, you may need one every once in a blue moon

However, even if we don't use IDs in CSS as much as we used to, they are still necessary

» Page fragment identifiers
» Unique hooks for JavaScript

IDs as page fragment identifiers

Given this HTML on `http://www.foo.com/toc/`:
`<h2 id="chapter2">`

You could link directly to it on the same page:
`<a href="#chapter2">`

Or from a different page:
`<a href="http://www.foo.com/toc/#chapter2">`

HTML — Tidy ✖

```html
1 ▾ <h3 id="toc">Table of Contents</h3>
2 ▾ <ul>
3 ▾   <li><a href="#the-tomb">The Tomb</a></li>
4 ▾   <li><a href="#dagon">Dagon</a></li>
5 ▾   <li><a href="#beyond-the-walls">Beyond the Walls
     of Sleep</a></li>
6   </ul>
7
8 ▾ <h3 id="the-tomb">The Tomb <a href="#toc">[TOC]</a>
     </h3>
9 ▾ <p>In relating the circumstances which have led to
     my confinement…</p>
10
11 ▾ <h3 id="dagon">Dagon <a href="#toc">[TOC]</a></h3>
12 ▾ <p>I am writing this under an appreciable mental
     strain…</p>
13
14 ▾ <h3 id="beyond-the-walls">Beyond the Walls of Sleep
     <a href="#toc">[TOC]</a></h3>
15 ▾ <p>I have frequently wondered if the majority of
     mankind…</p>
```

CSS — Tidy | View Compiled | ✖

JS — Tidy | ✖

Table of Contents

- The Tomb
- Dagon
- Beyond the Walls of Sleep

The Tomb   [TOC]

In relating the circumstances which have led to my confinement…

Dagon   [TOC]

I am writing this under an appreciable mental strain…

Beyond the Walls of Sleep   [TOC]

I have frequently wondered if the majority of mankind…

HTML | Tidy | ✕

```html
1 <p>
2   <a href="javascript:void(0)" onclick="toggle_show('answer');">
3   Toggle visibility
4 </a>
5 </p>
6 <p id="answer" style="display:none;">
7   Foo
8 </p>
```

CSS | Tidy | View Compiled | ✕

JS | Tidy | ✕

```javascript
1 function toggle_show(id) {
2   var e = document.getElementById(id);
3   if(e.style.display == 'block')
4     e.style.display = 'none';
5   else
6     e.style.display = 'block';
7 }
```

Toggle visibility

This parameter...

...helps the JavaScript...

...target this ID

HTML

Tidy ✕

```html
<p>
  <a href="javascript:void(0)" onclick="toggle_show('answer');">
  Toggle visibility
  </a>
</p>
<p id="answer" style="display:none;">
  Foo
</p>
```

CSS

Tidy   View Compiled   ✕

JS

Tidy ✕

```javascript
function toggle_show(id) {
  var e = document.getElementById(id);
  if(e.style.display == 'block')
    e.style.display = 'none';
  else
    e.style.display = 'block';
}
```

Toggle visibility

Foo

Other simple selectors

» Attribute selector
» Pseudo-class
» Pseudo-element

Those will be covered in *CSS - Selectors*

Compound Selectors

A *compound selector* describes *multiple conditions* on an element

If *all* conditions are true, the element is selected

Consists of a chain of simple selectors connected together, but *not* connected by a *combinator* (which is coming up next)

`p.alert` (usually `.alert` is all you want, allowing you to use the class with any element)

`table.inventory`

`table.inventory.northwest`

`.cthulhu:first-child`

A compound selector is used to match an element that has two classes on it

```
<p class="center note">
```

```
.center {
  text-align: center;
}
```
Selects elements with the .center class

```
.note {
  font-size: .9em;
}
```
Selects elements with the .note class

```
.center.note {
  font-style: italic;
}
```
Selects elements with the .center *and* .note classes

```
1 <p class="red">
2   Red
3 </p>
4 <p class="yellow">
5   Yellow
6 </p>
7 <p class="red yellow">
8   Orange
9 </p>
```

Red

Yellow

Orange

```
1  .red {
2    background-color: red;
3  }
4
5  .yellow {
6    background-color: yellow;
7  }
8
9  .red.yellow {
10   background-color: orange;
11 }
12
```

Complex Selectors Using Combinators

A *complex selector* uses *combinator(s)* to combine multiple selectors together into 1 selector

␣ (space)
\> (greater than)
\+ (plus)
~ (tilde)

Key Selectors

Rendering engines match selectors from right to left

The right-most part is the *key selector*, the actual object being selected

Really helpful understanding complex selectors

```
ul li

ul > li

ul > li a[title="home"]

.callOut > p:last-child

.ws-header .nav > li > a

p code, pre code, blockquote code
```

Descendant Combinator

```
selectorA selectorB
```

Selects *all* `selectorB` who have `selectorA` as *an ancestor*

`selectorB` can be a child, grandchild, or later descendant of `selectorA`

Any other `selectorB` who does *not* have `selectorA` as an ancestor is unaffected

```html
<ul class="lovecraft">
  <li>Great Old Ones
    <ul>
      <li>Cthugha</li>
      <li>Cthulhu</li>
      <li>Cyäegha</li>
    </ul>
  </li> <!-- end Great Old Ones -->
  <li>Gods
    <ul>
      <li>Outer Gods
        <ul>
          <li>Azathoth</li>
          <li>Nyarlathotep</li>
          <li>Yog-Sothoth</li>
        </ul>
      </li> <!-- end Outer Gods -->
      <li>Elder Gods
        <ul>
          <li>Bastet</li>
          <li>Hypnos</li>
          <li>Nodens</li>
        </ul>
      </li> <!-- end Elder Gods -->
    </ul>
```

CSS (SCSS)

JS

- Great Old Ones
  - Cthugha
  - Cthulhu
  - Cyäegha
- Gods
  - Outer Gods
    - Azathoth
    - Nyarlathotep
    - Yog-Sothoth
  - Elder Gods
    - Bastet
    - Hypnos
    - Nodens

Default for nested `<li>`: disc •, circle ◦, square ■

HTML

```html
<ul class="lovecraft">
  <li>Great Old Ones
    <ul>
      <li>Cthugha</li>
      <li>Cthulhu</li>
      <li>Cyäegha</li>
    </ul>
  </li> <!-- end Great Old Ones -->
  <li>Gods
    <ul>
      <li>Outer Gods
        <ul>
          <li>Azathoth</li>
          <li>Nyarlathotep</li>
          <li>Yog-Sothoth</li>
        </ul>
      </li> <!-- end Outer Gods -->
```

CSS (SCSS)

```scss
ul.lovecraft li {
  list-style-type: square;
}

/* Oh nos! I only wanted the <li> children of the top-
level <ul> selected, not all of the <li>'s! The perils
of the descendant combinator illustrated. */
```

JS

- Great Old Ones
  - Cthugha
  - Cthulhu
  - Cyäegha
- Gods
  - Outer Gods
    - Azathoth
    - Nyarlathotep
    - Yog-Sothoth
  - Elder Gods
    - Bastet
    - Hypnos
    - Nodens

I only wanted these `<li>`s to change to ■, not *all* of them

HTML

```html
<h2>The Terrible Old Man <small>
<a href="#toc">[TOC]</a></small>
</h2>

<p>
  It was the design of Angelo
Ricci and Joe Czanek and Manuel
Silva<span class="fn-symbol">*
</span> to call on the Terrible
Old Man. This old man dwells all
alone in a very ancient house on
Water Street near the sea, and is
reputed to be both exceedingly
rich and exceedingly feeble;
which forms a situation very
attractive to men of the
profession of Messrs. Ricci,
Czanek, and Silva, for that
profession was nothing less
dignified than robbery.
</p>
<p>
```

CSS

JS

The Terrible Old Man   [TOC]

It was the design of Angelo Ricci and Joe Czanek and Manuel Silva* to call on the Terrible Old Man. This old man dwells all alone in a very ancient house on Water Street near the sea, and is reputed to be both exceedingly rich and exceedingly feeble; which forms a situation very attractive to men of the profession of Messrs. Ricci, Czanek, and Silva, for that profession was nothing less dignified than robbery.

The inhabitants of Kingsport say and think many things about the Terrible Old Man which generally keep him safe from the attention of gentlemen like Mr. Ricci and his colleagues, despite the almost certain fact that he hides a fortune of indefinite magnitude somewhere about his musty and venerable abode.

* Note the "foreign" names that Lovecraft chooses.

The Terrible Old Man [TOC]

It was the design of Angelo Ricci and Joe Czanek and Manuel Silva* to call on the Terrible Old Man. This old man dwells all alone in a very ancient house on Water Street near the sea, and is reputed to be both exceedingly rich and exceedingly feeble; which forms a situation very attractive to men of the profession of Messrs. Ricci, Czanek, and Silva, for that profession was nothing less dignified than robbery.

The inhabitants of Kingsport say and think many things about the Terrible Old Man which generally keep him safe from the attention of gentlemen like Mr. Ricci and his colleagues, despite the almost certain fact that he hides a fortune of indefinite magnitude somewhere about his musty and venerable abode.

---

* Note the "foreign" names that Lovecraft chooses.

© 1921 H.P. Lovecraft

HTML

```html
10 ▾ <p>
11 ▾     <small><span class="fn-symbol">*
    </span> Note the "foreign" names
    that Lovecraft chooses.</small>
12      </p>
13      <footer>
```

CSS

```css
1 ▾ small {
2     font-size: .6em;
3   }
4
5 ▾ h2 small {
6     padding-left: 1em;
7   }
8
9 ▾ a {
10    text-decoration: none;
11  }
12
13 ▾ footer a {
14    border-bottom: 1px dotted white;
15    color: white;
16  }
17
```

JS

The Terrible Old Man   [TOC]

It was the design of Angelo Ricci and Joe Czanek and Manuel Silva* to call on the Terrible Old Man. This old man dwells all alone in a very ancient house on Water Street near the sea, and is reputed to be both exceedingly rich and exceedingly feeble; which forms a situation very attractive to men of the profession of Messrs. Ricci, Czanek, and Silva, for that profession was nothing less dignified than robbery.

The inhabitants of Kingsport say and think many things about the Terrible Old Man which generally keep him safe from the attention of gentlemen like Mr. Ricci and his colleagues, despite the almost certain fact that he hides a fortune of indefinite magnitude somewhere about his musty and venerable abode.

* Note the "foreign" names that Lovecraft chooses.

© 1921 H.P. Lovecraft

HTML

```html
<h2>The Terrible Old Man <small><a
href="#toc">[TOC]</a></small></h2>

<p>
    It was the design of Angelo
Ricci and Joe Czanek and Manuel
```

CSS

```css
small {
  font-size: .6em;
}

h2 small {
  padding-left: 1em;
}

a {
  text-decoration: none;
}

footer a {
  border-bottom: 1px dotted white;
  color: white;
}
```

JS

The Terrible Old Man   [TOC]

It was the design of Angelo Ricci and Joe Czanek and Manuel Silva* to call on the Terrible Old Man. This old man dwells all alone in a very ancient house on Water Street near the sea, and is reputed to be both exceedingly rich and exceedingly feeble; which forms a situation very attractive to men of the profession of Messrs. Ricci, Czanek, and Silva, for that profession was nothing less dignified than robbery.

The inhabitants of Kingsport say and think many things about the Terrible Old Man which generally keep him safe from the attention of gentlemen like Mr. Ricci and his colleagues, despite the almost certain fact that he hides a fortune of indefinite magnitude somewhere about his musty and venerable abode.

* Note the "foreign" names that Lovecraft chooses.

© 1921 H.P. Lovecraft

HTML

```html
<h2>The Terrible Old Man <small><a href="#toc">[TOC]</a></small></h2>

<p>
    It was the design of Angelo
Ricci and Joe Czanek and Manuel
```

CSS

```css
small {
  font-size: .6em;
}

h2 small {
  padding-left: 1em;
}

a {
  text-decoration: none;
}

footer a {
  border-bottom: 1px dotted white;
  color: white;
}
```

JS

The Terrible Old Man   [TOC]

It was the design of Angelo Ricci and Joe Czanek and Manuel Silva* to call on the Terrible Old Man. This old man dwells all alone in a very ancient house on Water Street near the sea, and is reputed to be both exceedingly rich and exceedingly feeble; which forms a situation very attractive to men of the profession of Messrs. Ricci, Czanek, and Silva, for that profession was nothing less dignified than robbery.

The inhabitants of Kingsport say and think many things about the Terrible Old Man which generally keep him safe from the attention of gentlemen like Mr. Ricci and his colleagues, despite the almost certain fact that he hides a fortune of indefinite magnitude somewhere about his musty and venerable abode.

* Note the "foreign" names that Lovecraft chooses.

© 1921 H.P. Lovecraft

HTML

```
13  <footer>
14    © 1921 <a
    href="http://www.hplovecraft.com/w
    ritings/fiction/tom.aspx">H.P.
    Lovecraft</a>
15  </footer>
```

CSS

```
1   small {
2     font-size: .6em;
3   }
4
5   h2 small {
6     padding-left: 1em;
7   }
8
9   a {
10    text-decoration: none;
11  }
12
13  footer a {
14    border-bottom: 1px dotted white;
15    color: white;
16  }
17
```

JS

The Terrible Old Man   [TOC]

It was the design of Angelo Ricci and Joe Czanek and Manuel Silva* to call on the Terrible Old Man. This old man dwells all alone in a very ancient house on Water Street near the sea, and is reputed to be both exceedingly rich and exceedingly feeble; which forms a situation very attractive to men of the profession of Messrs. Ricci, Czanek, and Silva, for that profession was nothing less dignified than robbery.

The inhabitants of Kingsport say and think many things about the Terrible Old Man which generally keep him safe from the attention of gentlemen like Mr. Ricci and his colleagues, despite the almost certain fact that he hides a fortune of indefinite magnitude somewhere about his musty and venerable abode.

---

* Note the "foreign" names that Lovecraft chooses.

© 1921 H.P. Lovecraft

You can often use a descendant combinator instead of a class (& you must if you cannot change the HTML)

👎

HTML
```
<aside>
  <img
class="headshot"
src="…">
</aside>
```
CSS
```
.headshot {}
```

👍

Cleaner HTML
```
<aside>
  <img src="…">
</aside>
```

Better CSS using the descendant combinator
```
aside img {}
```

Child Combinator

```
selectorA > selectorB
```

Selects any `selectorB` who is a *direct child* of `selectorA`, not a grandchild or any other descendants

All siblings who are direct children of `selectorA` are selected
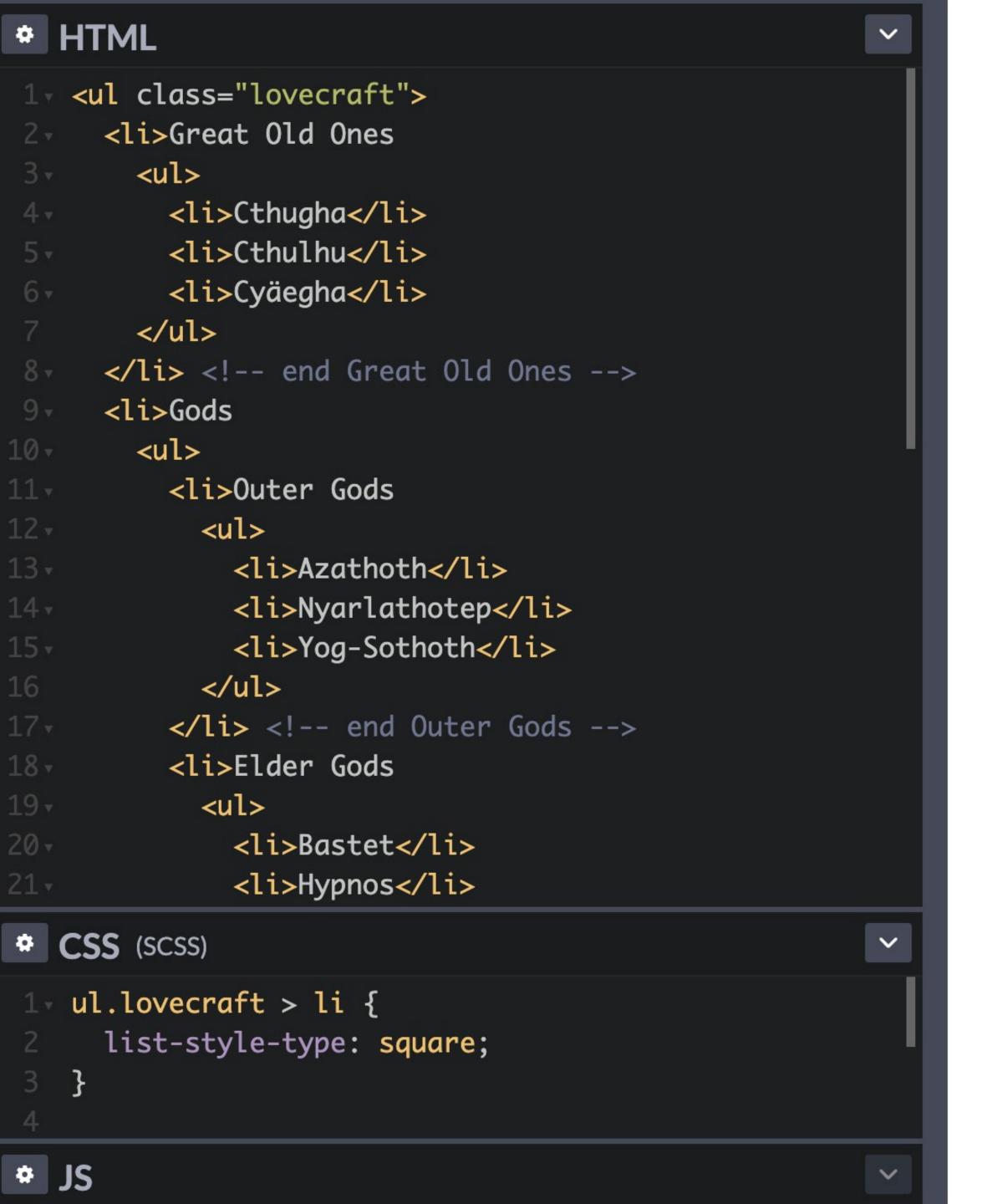
Siblings: 2 or more elements that share a parent
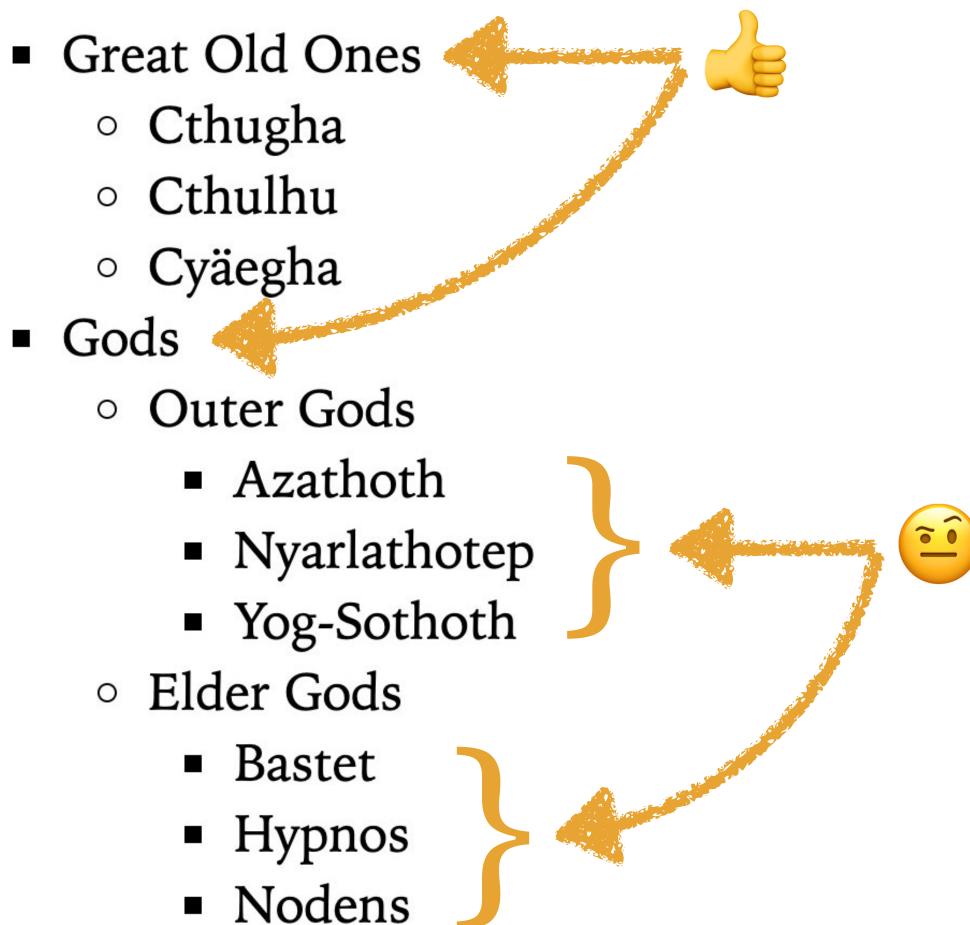
Contrasts with the descendant combinator, which selects both direct children & any descendants, no matter how deep

HTML

```html
<ul class="lovecraft">
  <li>Great Old Ones
    <ul>
      <li>Cthugha</li>
      <li>Cthulhu</li>
      <li>Cyäegha</li>
    </ul>
  </li> <!-- end Great Old Ones -->
  <li>Gods
    <ul>
      <li>Outer Gods
        <ul>
          <li>Azathoth</li>
          <li>Nyarlathotep</li>
          <li>Yog-Sothoth</li>
        </ul>
      </li> <!-- end Outer Gods -->
      <li>Elder Gods
        <ul>
          <li>Bastet</li>
          <li>Hypnos</li>
          <li>Nodens</li>
        </ul>
      </li> <!-- end Elder Gods -->
    </ul>
```

CSS (SCSS)

JS

- Great Old Ones
  - Cthugha
  - Cthulhu
  - Cyäegha
- Gods
  - Outer Gods
    - Azathoth
    - Nyarlathotep
    - Yog-Sothoth
  - Elder Gods
    - Bastet
    - Hypnos
    - Nodens

Default for nested `<li>`: disc •, circle ○, square ■

HTML

```html
<ul class="lovecraft">
  <li>Great Old Ones
    <ul>
      <li>Cthugha</li>
      <li>Cthulhu</li>
      <li>Cyäegha</li>
    </ul>
  </li> <!-- end Great Old Ones -->
  <li>Gods
    <ul>
      <li>Outer Gods
        <ul>
          <li>Azathoth</li>
          <li>Nyarlathotep</li>
          <li>Yog-Sothoth</li>
        </ul>
      </li> <!-- end Outer Gods -->
      <li>Elder Gods
        <ul>
          <li>Bastet</li>
          <li>Hypnos</li>
```

CSS (SCSS)

```scss
ul.lovecraft > li {
  list-style-type: square;
}
```

JS

- Great Old Ones 👍
  - Cthugha
  - Cthulhu
  - Cyäegha
- Gods
  - Outer Gods
    - Azathoth
    - Nyarlathotep 🤨
    - Yog-Sothoth
  - Elder Gods
    - Bastet
    - Hypnos
    - Nodens

1st level changed from • to ▪, but 3rd level remains ▪ because that's the default

```html
       <li>Cthugha</li>
       <li>Cthulhu</li>
       <li>Cyäegha</li>
     </ul>
   </li> <!-- end Great Old Ones -->
   <li>Gods
     <ul>
       <li>Outer Gods
         <ul>
           <li>Azathoth</li>
           <li>Nyarlathotep</li>
           <li>Yog-Sothoth</li>
         </ul>
       </li> <!-- end Outer Gods -->
       <li>Elder Gods
```

CSS (SCSS) `Compiled`

```scss
ul.lovecraft > li {
  list-style-type: square;
}
ul.lovecraft > li > ul > li {
  list-style-type: disc;
}
ul.lovecraft > li > ul > li > ul > li {
  list-style-type: circle;
}
```

JS

- Great Old Ones
  - Cthugha
  - Cthulhu
  - Cyäegha
- Gods
  - Outer Gods
    - Azathoth
    - Nyarlathotep
    - Yog-Sothoth
  - Elder Gods
    - Bastet
    - Hypnos
    - Nodens

Let's change all 3 level defaults: 1st from • to ■, 2nd from ○ to •, & 3rd from ■ to ○

```html
<table class="movies">
  <thead>
    <tr>
      <th>Movie</th>
      <th>Genre</th>
      <th>Hero</th>
      <th>Year</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Die Hard</td>
      <td>Action</td>
      <td>John McClane</td>
      <td>1988</td>
    </tr>
    <tr>
      <td>Star Wars</td>
      <td>Sci-Fi</td>
      <td>Luke Skywalker</td>
      <td>1977</td>
    </tr>
    <tr>
      <td>John Wick</td>
      <td>Action</td>
```

CSS

JS

Movie	Genre	Hero	Year
Die Hard	Action	John McClane	1988
Star Wars	Sci-Fi	Luke Skywalker	1977
John Wick	Action	John Wick	2014

```
HTML

1  <table class="movies">
2    <thead>↔</thead>
10   <tbody>
11     <tr>
12       <td>Die Hard</td>
13       <td>Action</td>
14       <td>John McClane</td>
15       <td>1988</td>
16     </tr>
17     <tr>
18       <td>Star Wars</td>
19       <td>Sci-Fi</td>
20       <td>Luke Skywalker</td>
21       <td>1977</td>
22     </tr>
23     <tr>
24       <td>John Wick</td>
25       <td>Action</td>
26       <td>John Wick</td>
27       <td>2014</td>
28     </tr>
29   </tbody>
30 </table>
```

Movie	Genre	Hero	Year
Die Hard	Action	John McClane	1988
Star Wars	Sci-Fi	Luke Skywalker	1977
John Wick	Action	John Wick	2014

CSS

JS

HTML

```
1  <table class="movies">
2    <thead>
3      <tr>
4        <th>Movie</th>
5        <th>Genre</th>
6        <th>Hero</th>
7        <th>Year</th>
8      </tr>
9    </thead>
10   <tbody>
11     <tr>
12       <td>Die Hard</td>
13       <td>Action</td>
14       <td>John McClane</td>
15       <td>1988</td>
16     </tr>
17     <tr>
18       <td>Star Wars</td>
19       <td>Sci-Fi</td>
20       <td>Luke Skywalker</td>
21       <td>1977</td>
```

CSS

```
1  table.movies > tbody > tr:hover {
2    background-color: pink;
3  }
```

JS

Movie	Genre	Hero	Year
Die Hard	Action	John McClane	1988
Star Wars	Sci-Fi	Luke Skywalker	1977
John Wick	Action	John Wick	2014

HTML

```html
<table class="movies">
  <thead>
    <tr>
      <th>Movie</th>
      <th>Genre</th>
      <th>Hero</th>
      <th>Year</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Die Hard</td>
      <td>Action</td>
      <td>John McClane</td>
      <td>1988</td>
    </tr>
    <tr>
      <td>Star Wars</td>
      <td>Sci-Fi</td>
      <td>Luke Skywalker</td>
      <td>1977</td>
```

CSS

```css
table.movies > tbody > tr:hover {
  background-color: pink;
}
```

JS

Movie	Genre	Hero	Year
Die Hard	Action	John McClane	1988
Star Wars	Sci-Fi	Luke Skywalker	1977
John Wick	Action	John Wick	2014

An illustration of the difference between the descendant & child combinators

HTML

```
1 ▾ <div>
2 ▸   <p>↔</p>
5 ▾   <blockquote>
6 ▸     <p>↔</p>
9 ▸     <p>↔</p>
12    </blockquote>
13 ▸   <p>↔</p>
16  </div>
17
18 ▾ <!-- From H.P. Lovecraft's "The
   Tomb" at
   http://www.hplovecraft.com
   /writings/texts/fiction/t.aspx
   -->
```

CSS

```
1 ▾ /* div p:last-child { */
2 ▾ div > p:last-child {
3     font-family: Creepster,
   fantasy;
4     font-size: 1.2em;
5 }
6
```

JS

On the following day I was brought to this room with the barred windows, but I have been kept informed of certain things through an aged and simple-minded servitor, for whom I bore a fondness in infancy, and who like me loves the churchyard.

> What I have dared relate of my experiences within the vault has brought me only pitying smiles. My father, who visits me frequently, declares that at no time did I pass the chained portal, and swears that the rusted padlock had not been touched for fifty years when he examined it.
>
> He even says that all the village knew of my journeys to the tomb, and that I was often watched as I slept in the bower outside the grim facade, my half-open eyes fixed on the crevice that leads to the interior.

AGAINST THESE ASSERTIONS I HAVE NO TANGIBLE PROOF TO OFFER, SINCE MY KEY TO THE PADLOCK WAS LOST IN THE STRUGGLE ON THAT NIGHT OF HORRORS. THE STRANGE THINGS OF THE PAST WHICH I LEARNT DURING THOSE NOCTURNAL MEETINGS WITH THE DEAD HE DISMISSES AS THE FRUITS OF MY LIFELONG AND OMNIVOROUS BROWSING AMONGST THE ANCIENT VOLUMES OF THE FAMILY LIBRARY.

Using > limits the scope of the styles

HTML

```html
<div>
  <p>↔</p>
  <blockquote>
    <p>↔</p>
    <p>↔</p>
  </blockquote>
  <p>↔</p>
</div>

<!-- From H.P. Lovecraft's "The
Tomb" at
http://www.hplovecraft.com
/writings/texts/fiction/t.aspx
-->
```

CSS

```css
div p:last-child {
/* div > p:last-child { */
  font-family: Creepster,
fantasy;
  font-size: 1.2em;
}
```

JS

On the following day I was brought to this room with the barred windows, but I have been kept informed of certain things through an aged and simple-minded servitor, for whom I bore a fondness in infancy, and who like me loves the churchyard.

What I have dared relate of my experiences within the vault has brought me only pitying smiles. My father, who visits me frequently, declares that at no time did I pass the chained portal, and swears that the rusted padlock had not been touched for fifty years when he examined it.

HE EVEN SAYS THAT ALL THE VILLAGE KNEW OF MY JOURNEYS TO THE TOMB, AND THAT I WAS OFTEN WATCHED AS I SLEPT IN THE BOWER OUTSIDE THE GRIM FACADE, MY HALF-OPEN EYES FIXED ON THE CREVICE THAT LEADS TO THE INTERIOR.

AGAINST THESE ASSERTIONS I HAVE NO TANGIBLE PROOF TO OFFER, SINCE MY KEY TO THE PADLOCK WAS LOST IN THE STRUGGLE ON THAT NIGHT OF HORRORS. THE STRANGE THINGS OF THE PAST WHICH I LEARNT DURING THOSE NOCTURNAL MEETINGS WITH THE DEAD HE DISMISSES AS THE FRUITS OF MY LIFELONG AND OMNIVOROUS BROWSING AMONGST THE ANCIENT VOLUMES OF THE FAMILY LIBRARY.

Selector List

`selectorA, selectorB, selectorC`

List selectors that have similar declarations for simpler & cleaner CSS & HTML

Don't do this:

```css
p {
  font-family: Verdana, sans-serif;
  font-size: 1em;
}


blockquote {
  font-family: Verdana, sans-serif;
  font-size: 1em;
}
```

Do this:

```css
blockquote, p {
  font-family: Verdana, sans-serif;
  font-size: 1em;
}
```

Any selector can be included in the list

```
.emphasis, .title {font-style: italic;}

em, .title {font-style: italic;}

p, #nav, .pullquote {font-family: Verdana,
sans-serif;} 🤪
```

A very common selector list

```
h1, h2, h3, h4, h5, h6 {
    …
}
```

Group similar selectors, but be specific where needed

CSS is cumulative unless overridden

```css
blockquote, p {
  font-family: Verdana, sans-serif;
}


p {
  line-height: 1.5;
}
```

Turn this:

```
h1 {
  font-weight: normal;
  font-size: 2.5em;
  font-family: serif;
}

h2 {
  font-weight: normal;
  border-bottom: 1px
dotted black;
  font-family: serif;
  font-size: 1.8em;
}
```

Into this:

```
h1, h2 {
  font-weight: normal;
  font-family: serif;
}

h1 {
  font-size: 2.5em;
}

h2 {
  border-bottom: 1px dotted
black;
  font-size: 1.8em;
}
```

Good practice

```
blockquote,
option,
p,
td,
#sidebar,
.legalese {
  font-family: Verdana, sans-serif;
  font-size: 1em;
}
```

Elements, then IDs, then classes

Alphabetical order within each grouping of selectors

Formatting

Don't do this:

```css
h1 {color: dimgray;}
h1 {font-size: 1.4em;}
h1 {font-weight: bold;}
h1 {font-family: Verdana, sans-serif;}
```

Instead, combine related declarations

```css
h1 {
  color: dimgray;
  font-family: Verdana, sans-serif;
  font-size: 1.4em;
  font-weight: bold;
}
```

Formatting CSS rulesets

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

The order of declarations in the declaration block doesn't matter

Do not forget to put ; at the end of every line in a ruleset

You do not actually have to put ; at the end of the last line in a ruleset, but that is a very bad habit to get into

Use comments in CSS for the same reasons as in HTML

» *Notes* to yourself & others
» ~~*Debugging*: comment out troublesome CSS for testing~~ (use your browser's Inspector)

HTML comments
```
<!-- blah blah html blah blah html -->
```

CSS comments
```
/* blah blah css css blah blah css css */
```
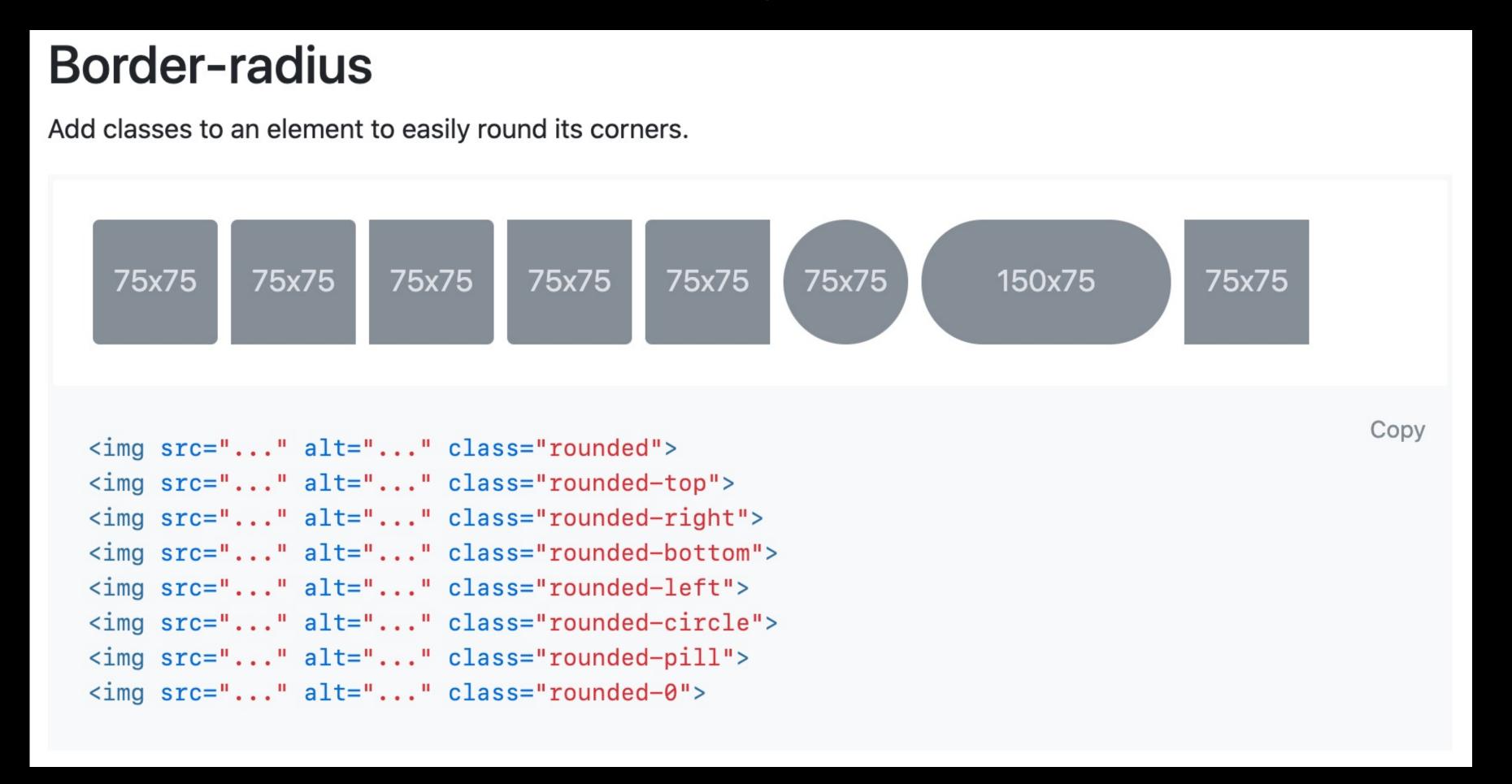
Design Pattern

"a formal way of documenting a solution to a design problem in a particular field of expertise." —Wikipedia

"Each pattern describes a problem that occurs over and over again in our environment, and then describes the core of the solution to that problem" —Christopher Alexander, architect & author of *A Pattern Language* (1977)

Border-radius

Add classes to an element to easily round its corners.

| 75x75 | 75x75 | 75x75 | 75x75 | 75x75 | 75x75 | 150x75 | 75x75 |

Copy

```
<img src="..." alt="..." class="rounded">
<img src="..." alt="..." class="rounded-top">
<img src="..." alt="..." class="rounded-right">
<img src="..." alt="..." class="rounded-bottom">
<img src="..." alt="..." class="rounded-left">
<img src="..." alt="..." class="rounded-circle">
<img src="..." alt="..." class="rounded-pill">
<img src="..." alt="..." class="rounded-0">
```

Bootstrap 4's classes for the common design pattern of rounded borders

📝 SIDE NOTE


```html
<div class="card" style="width: 18rem;">
  <img src="..." class="card-img-top" alt="...">
  <div class="card-body">
    <h5 class="card-title">Card title</h5>
    <p class="card-text">Some quick example text to build on the card title and make up the
    <a href="#" class="btn btn-primary">Go somewhere</a>
  </div>
</div>
```
Copy

Bootstrap 4's classes for the common design pattern of cards

Here's the order Jans normally uses in his stylesheets

1. General rules that apply to the whole site (linked libraries, typography)
2. Site-wide design patterns (header, nav, footer)
3. Page type design patterns (sidebar, news, post)
4. Specific page design patterns (home page)
5. Content design patterns (callouts, image gallery)

With appropriate comments sprinkled throughout

Here's the order I sometimes shoot for in my stylesheets

```
@font-face
html
body
/* General */
<type selectors, A→Z>
<ID selectors, A→Z>
<class selectors, A→Z>
/* <New Section> */
<type>
<ID>
<class>
…
```

<span>
&
<div>

HTML elements "work" without attributes & values

(with a tiny few exceptions, like `<img>` & `<a>`)

`<span>` & `<div>` are HTML elements designed solely to work with CSS*

`<span>` & `<div>` by themselves do nothing on a webpage (except draw invisible boxes)

They must use CSS (`class="foo"` or `id="bar"`) to do anything productive

* & JavaScript

`<span>`

`<span>` is a text semantic element that creates an *inline box* & does nothing else without CSS

Use `<span>` to hold attribute-value pairs relevant to CSS

As a text semantic element, it usually doesn't have other elements nested inside it

```
<span class="foo">Ph'nglui mglw'nafh Cthulhu R'lyeh
wgah'nagl fhtagn!</span>
```

Use it when other text semantic elements are not semantically appropriate

HTML

```html
4 <p>
5   I heard it, and knew no more. Heard it
    as I sat petrified in that unknown
    cemetery in the hollow, amidst the
    crumbling stones and the falling
    tombs, the rank vegetation and the
    miasmal vapours. Heard it well up from
    the innermost depths of that damnable
    open sepulchre as I watched amorphous,
    necrophagous shadows dance beneath an
    accursed waning moon. And this is what
    it said: "<span class="eeek">You fool,
    Warren is dead!</span>"
6 </p>
```

CSS

```css
1  .eeek {
2    text-transform: uppercase;
3    font-family: Creepster, cursive;
4    font-size: 110%;
5  }
6
7  /* Uninteresting stuff below */
8
9  p {
10   font-size: 150%;
11   font-family: "Iowan Old Style",
   Georgia, serif;
```

JS

From H. P. Lovecraft's "The Statement of Randolph Carter":

I heard it, and knew no more. Heard it as I sat petrified in that unknown cemetery in the hollow, amidst the crumbling stones and the falling tombs, the rank vegetation and the miasmal vapours. Heard it well up from the innermost depths of that damnable open sepulchre as I watched amorphous, necrophagous shadows dance beneath an accursed waning moon. And this is what it said: "YOU FOOL, WARREN IS DEAD!"
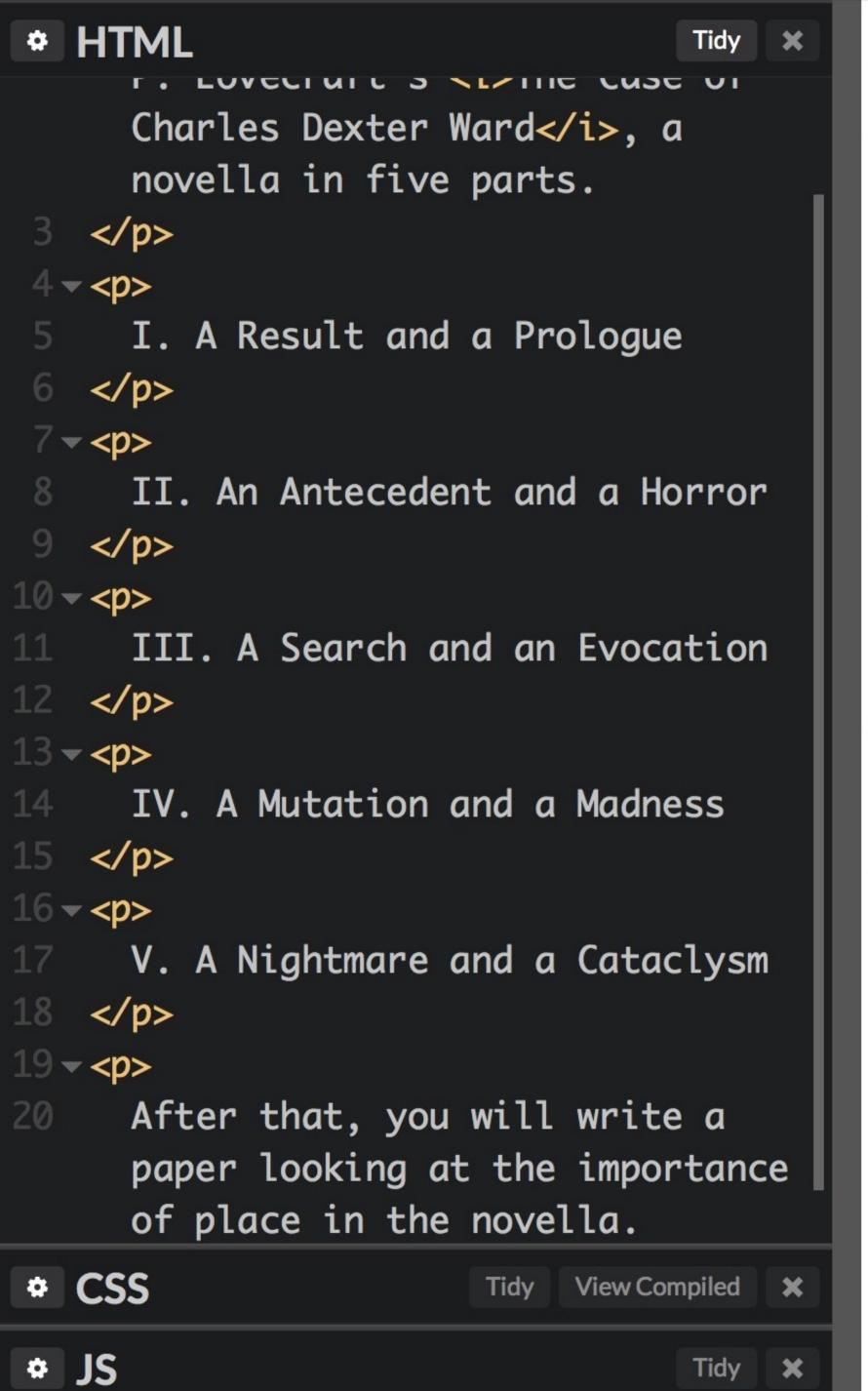
`<div>`

`<div>` is a grouping element that creates a *block box* by default & does nothing else without CSS

Use `<div>` to hold attribute-value pairs relevant to CSS

As a grouping element, it groups other elements, e.g., put a `<div>` around 3 `<p>`s & a `<ul>`

Use it when other grouping elements are not semantically appropriate

```
   P. Lovecraft's <i>The Case of
   Charles Dexter Ward</i>, a
   novella in five parts.
3  </p>
4 ▾ <p>
5    I. A Result and a Prologue
6  </p>
7 ▾ <p>
8    II. An Antecedent and a Horror
9  </p>
10 ▾ <p>
11   III. A Search and an Evocation
12  </p>
13 ▾ <p>
14   IV. A Mutation and a Madness
15  </p>
16 ▾ <p>
17   V. A Nightmare and a Cataclysm
18  </p>
19 ▾ <p>
20   After that, you will write a
     paper looking at the importance
     of place in the novella.
```

Your assignment is to read H. P. Lovecraft's *The Case of Charles Dexter Ward,* a novella in five parts.

I. A Result and a Prologue

II. An Antecedent and a Horror

III. A Search and an Evocation

IV. A Mutation and a Madness

V. A Nightmare and a Cataclysm

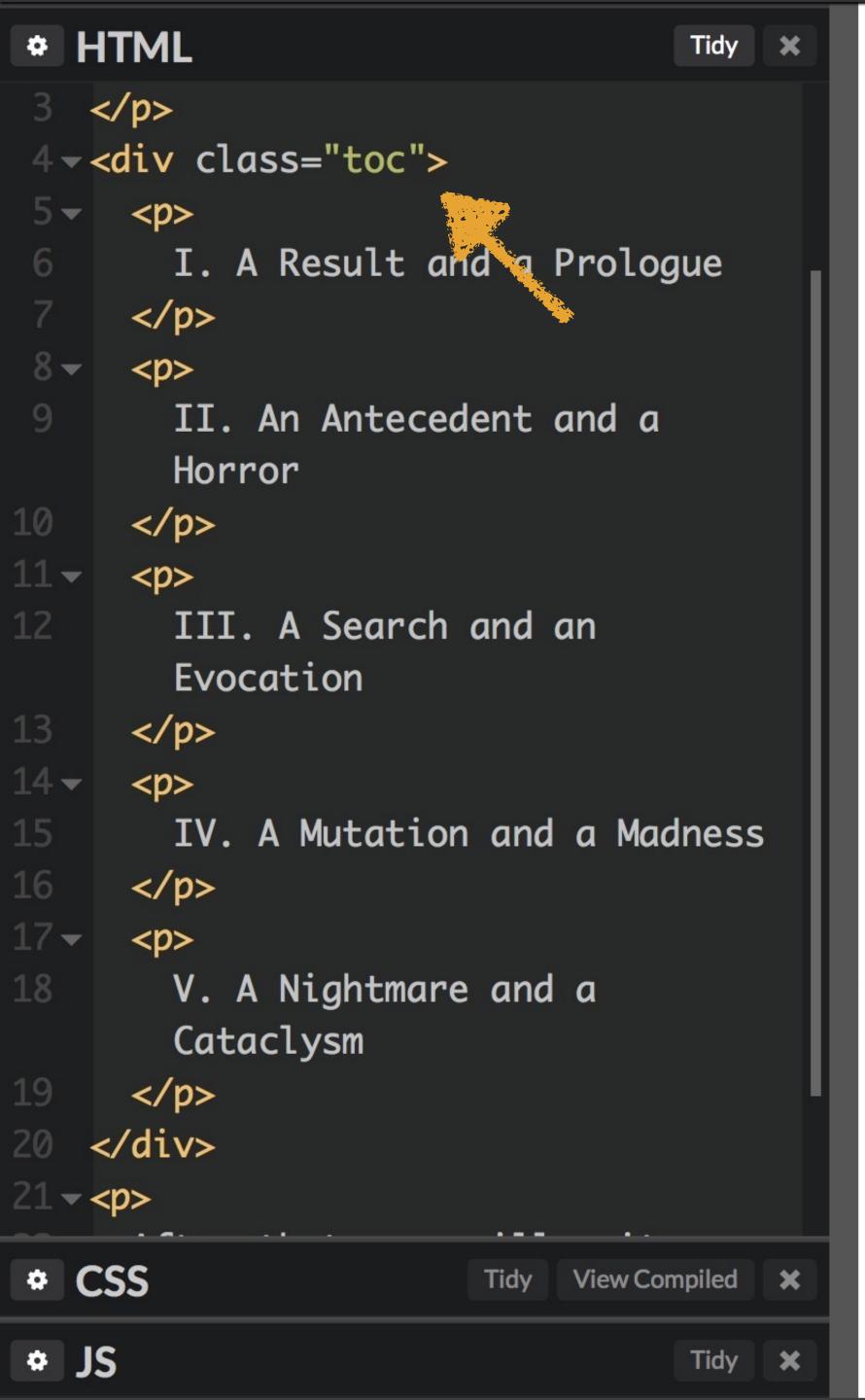After that, you will write a paper looking at the importance of place in the novella.

I want the table of contents to stand out with a background color, borders, & rounded corners

HTML
Tidy ✕

```html
1 <p>
2   Your assignment is to read H.
    P. Lovecraft's <i>The Case of
    Charles Dexter Ward</i>, a
    novella in five parts.
3 </p>
4 <p class="toc">
5   I. A Result and a Prologue
6 </p>
7 <p class="toc">
8   II. An Antecedent and a Horror
9 </p>
10 <p class="toc">
11   III. A Search and an Evocation
```

CSS
Tidy | View Compiled | ✕

```css
1 .toc {
2   background-color: #E8E8E8;
3   margin: 1em;
4   border: 1px solid #ccc;
5   padding: 0 1em;
6   width: 90%;
7   border-radius: 1em;
8 }
```

JS
Tidy ✕

Your assignment is to read H. P. Lovecraft's *The Case of Charles Dexter Ward,* a novella in five parts.

> I. A Result and a Prologue

> II. An Antecedent and a Horror

> III. A Search and an Evocation

> IV. A Mutation and a Madness

> V. A Nightmare and a Cataclysm

After that, you will write a paper looking at the importance of place in the novella.

Well that looks stupid...

Your assignment is to read H. P. Lovecraft's *The Case of Charles Dexter Ward,* a novella in five parts.

> I. A Result and a Prologue
>
> II. An Antecedent and a Horror
>
> III. A Search and an Evocation
>
> IV. A Mutation and a Madness
>
> V. A Nightmare and a Cataclysm

After that, you will write a paper looking at the importance of place in the novella.
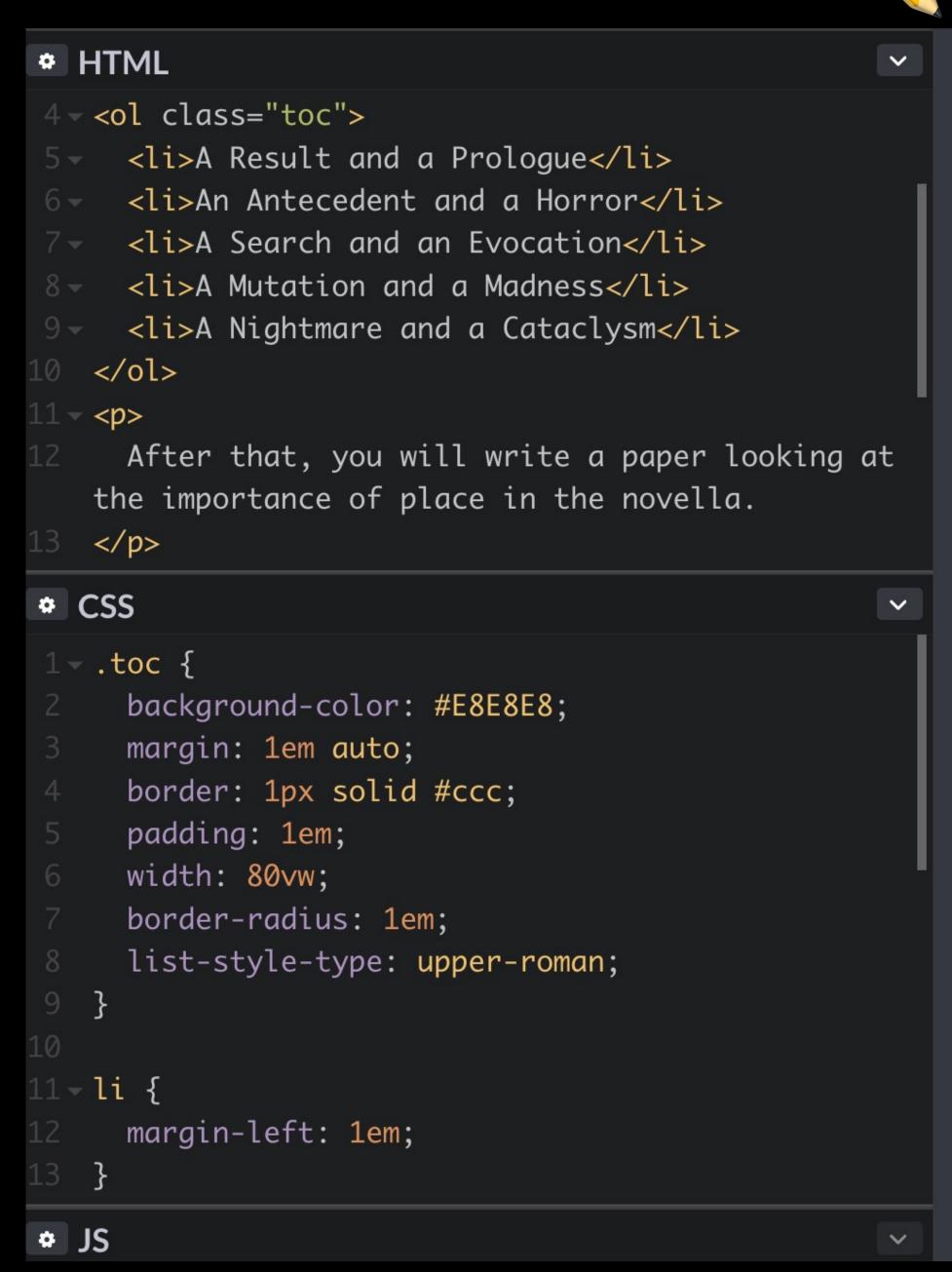

HTML Tidy ✖

```
 3    </p>
 4  ▾ <div class="toc">
 5  ▾   <p>
 6        I. A Result and a Prologue
 7      </p>
 8  ▾   <p>
 9        II. An Antecedent and a
          Horror
10      </p>
11  ▾   <p>
12        III. A Search and an
          Evocation
13      </p>
14  ▾   <p>
15        IV. A Mutation and a Madness
16      </p>
17  ▾   <p>
18        V. A Nightmare and a
          Cataclysm
19      </p>
20    </div>
21  ▾ <p>
```

CSS Tidy View Compiled ✖

JS Tidy ✖

HTML
Tidy

```html
1  <p>
2    Your assignment is to read H.
     P. Lovecraft's <i>The Case of
     Charles Dexter Ward</i>, a
     novella in five parts.
3  </p>
4  <div class="toc">
5    <p>
6      I. A Result and a Prologue
7    </p>
8    <p>
9      II. An Antecedent and a
       Horror
10   </p>
```

CSS
Tidy   View Compiled

```css
1  .toc {
2    background-color: #E8E8E8;
3    margin: 1em;
4    border: 1px solid #ccc;
5    padding: 0 1em;
6    width: 90%;
7    border-radius: 1em;
8  }
```

JS
Tidy

Your assignment is to read H. P. Lovecraft's *The Case of Charles Dexter Ward*, a novella in five parts.

> I. A Result and a Prologue
>
> II. An Antecedent and a Horror
>
> III. A Search and an Evocation
>
> IV. A Mutation and a Madness
>
> V. A Nightmare and a Cataclysm

After that, you will write a paper looking at the importance of place in the novella.

Much better—& that is why we have `<div>`

HTML

```
4   <ol class="toc">
5     <li>A Result and a Prologue</li>
6     <li>An Antecedent and a Horror</li>
7     <li>A Search and an Evocation</li>
8     <li>A Mutation and a Madness</li>
9     <li>A Nightmare and a Cataclysm</li>
10  </ol>
11  <p>
12    After that, you will write a paper looking at
      the importance of place in the novella.
13  </p>
```

CSS

```
1   .toc {
2     background-color: #E8E8E8;
3     margin: 1em auto;
4     border: 1px solid #ccc;
5     padding: 1em;
6     width: 80vw;
7     border-radius: 1em;
8     list-style-type: upper-roman;
9   }
10
11  li {
12    margin-left: 1em;
13  }
```

JS

Your assignment is to read H. P. Lovecraft's *The Case of Charles Dexter Ward,* a novella in five parts.

> I. A Result and a Prologue
> II. An Antecedent and a Horror
> III. A Search and an Evocation
> IV. A Mutation and a Madness
> V. A Nightmare and a Cataclysm

After that, you will write a paper looking at the importance of place in the novella.

To be semantic, I should really use an `<ol>`

This is a useless `<div>`

```
<div class="lead-copy">
  <p>
    When a traveller in north central Massachusetts takes
    the wrong fork at the junction of the Aylesbury pike just
    beyond Dean's Corners he comes upon a lonely and
    curious country.
  </p>
</div>
```

Only use `<div>` around 2 or more elements that create block boxes*

* It's OK to wrap a `<div>` around 1 element in a few cases

Inheritance

HTML

```
1 <p>
2   And they called him <span class="alert">Seamus</span>…
3 </p>
```

CSS

```
1  html {
2    font-size: 24px; /* inherited */
3  }
4
5  p {
6    font-family: Georgia, serif; /* inherited */
7  }
8
9  .alert {
10   color: red;
11 }
```

JS

And they called him Seamus...

Some properties, like `font-size` & `color`, are *inherited*: elements with those properties pass those properties down through the DOM to their descendant elements (unless overridden)

Other properties, like `background-image` & `border`, are *not inherited*: elements with those properties do not pass those properties down to their descendent elements

Inheritance is for elements that do not have properties set

HTML

```html
<blockquote>
  <p>
    Of the five remaining pictures, three were of
    swamp and hill scenes which seemed to bear
    traces of hidden and <em>unwholesome</em>
    tenancy. Another was of a queer mark in the
    ground very near Akeley's house…
  </p>
</blockquote>
```

CSS

```css
html {
    font-size: 24px; /* inherited */
}

blockquote {
    font-family: "Iowan Old Style", Georgia,
  serif; /* inherited */
    border: 1px solid black; /* not inherited */
    padding: 0 1em; /* not inherited */
}

```

Of the five remaining pictures, three were of swamp and hill scenes which seemed to bear traces of hidden and *unwholesome* tenancy. Another was of a queer mark in the ground very near Akeley's house…

JS

border-collapse
border-spacing
caption-side
color
cursor
direction
empty-cells
font-family
font-size
font-weight

font-style
font-variant
font
letter-spacing
list-style-type
list-style-position
list-style-image
list-style

line-height
orphans
quotes
text-align
text-edge
text-indent
text-transform
visibility
white-space
widows
word-spacing

Partial list of inherited properties

Cascading
Style Sheets

How does the rendering engine know which style to apply to an element?

If a selector matches an element, that selector's styles are used

Inheritance comes into play if an element does not have properties set

But what if the CSS rules *conflict*; e.g., what if CSS tells the rendering engine to make all `<p>`s use serif *and* sans-serif fonts?

If an element's CSS declaration conflicts with another declaration, the rendering engine uses the *Cascade* to find a winner

In other words, conflicting declarations follow a *cascade*, & the rule with the most *weight* wins

3 parts to the Cascade

1. Importance

2. Specificity

3. Order

Importance

CSS can come from 3 places:

» Browser, aka, the *user agent*
» User
» Author

All Web browsers have built-in CSS rules

In Firefox, for example, why does `<p>` have a certain amount of space before & after it?

Because of Firefox's built-in default CSS

layout/style/html.css

author	ffxbld
	Sat, 26 Dec 2015 04:05:31 -0800
changeset 277655	c39a71cf4146
parent 275624	ddf13e6aa97e
permissions	-rw-r--r--

No bug, Automated blocklist update from host bld-linux64-spot-593 - a=blocklist-update

```
1   /* This Source Code Form is subject to the terms of the Mozilla Public
2    * License, v. 2.0. If a copy of the MPL was not distributed with this
3    * file, You can obtain one at http://mozilla.org/MPL/2.0/. */
4
5   @namespace url(http://www.w3.org/1999/xhtml); /* set default namespace to HTML */
6   @namespace xul url(http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul);
7
8   /* bidi */
9
10  [dir="rtl"] {
11    direction: rtl;
12    unicode-bidi: embed;
13  }
14  [dir="ltr"] {
15    direction: ltr;
16    unicode-bidi: embed;
17  }
18
19  bdi:-moz-dir(ltr), [dir="auto"]:-moz-dir(ltr) { direction: ltr; }
20  bdi:-moz-dir(rtl), [dir="auto"]:-moz-dir(rtl) { direction: rtl; }
21
22  /* To ensure http://www.w3.org/TR/REC-html40/struct/dirlang.html#style-bidi:
23   *
24   * "When a block element that does not have a dir attribute is transformed to
25   * the style of an inline element by a style sheet, the resulting presentation
26   * should be equivalent, in terms of bidirectional formatting, to the
27   * formatting obtained by explicitly adding a dir attribute (assigned the
28   * inherited value) to the transformed element."
29   *
```

```
108  header,
109  hgroup,
110  html,
111  main,
112  nav,
113  section {
114      display: block;
115  }
116
117  body {
118      display: block;
119      margin: 8px;
120  }
121
122  p, dl, multicol {
123      display: block;
124      margin-block-start: 1em;
125      margin-block-end: 1em;
126  }
127
128  dd {
129      display: block;
130      -moz-margin-start: 40px;
131  }
132
133  blockquote, figure {
134      display: block;
135      margin-block-start: 1em;
136      margin-block-end: 1em;
137      -moz-margin-start: 40px;
138      -moz-margin-end: 40px;
139  }
140
141  address {
142      display: block;
```

Equivalent to `margin-top: 1em` & `margin-bottom: 1em`

WebKit (Safari)

source: trunk / Source / WebCore / css / html.css @ 194427

Visit: ⌄   View revision:

Revision 191751, 30.6 KB checked in by wenson_hsieh@apple.com, 2 months ago (diff)

Date input values should not overlap with menu list dropdown button on iOS
⇨ https://bugs.webkit.org/show_bug.cgi?id=150681
<rdar://problem/19965078>
Reviewed by Zalan Bujtas.
Source/WebCore:
Adds a right margin on iOS date and time inputs so that the inner div does
not render the value of the date on top of the dropdown button on the right
of the menu list.
Tests: fast/forms/date/date-input-rendering-basic.html

        fast/forms/time/time-input-rendering-basic.html

  • css/html.css:
(input::-webkit-date-and-time-value):
LayoutTests:
Tests that date and time inputs render properly. Namely, the text should not
be rendered such that it covers the dropdown button on the right of the menu
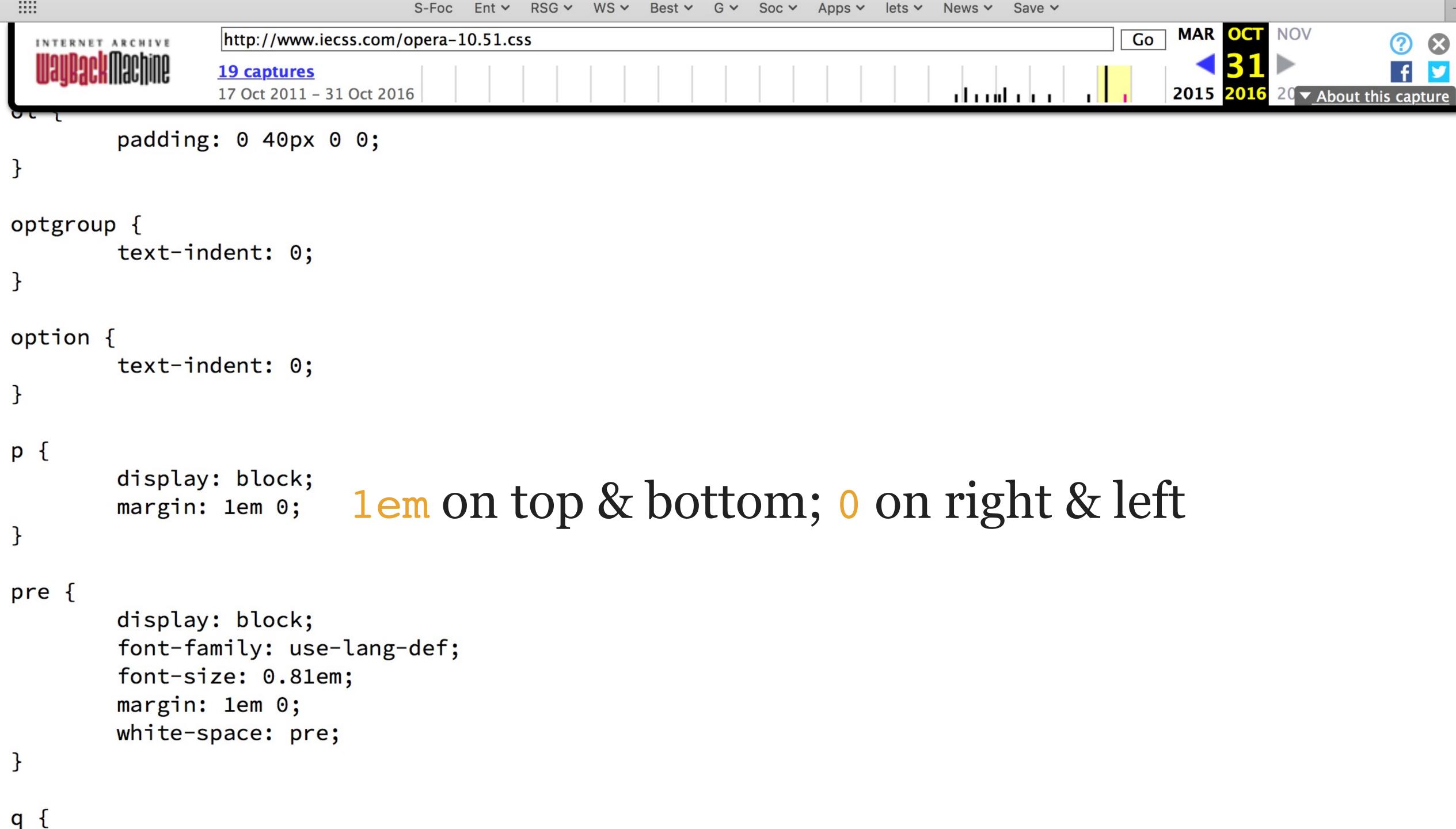list.
  • fast/forms/date/date-input-rendering-basic-expected.txt: Added.
  • fast/forms/date/date-input-rendering-basic.html: Added.
  • fast/forms/time/time-input-rendering-basic-expected.txt: Added.
  • fast/forms/time/time-input-rendering-basic.html: Added.

    Property **svn:eol-style** set to *native*

Line	
1	/*
2	* The default style sheet used to render HTML.
3	*
4	* Copyright (C) 2000 Lars Knoll (knoll@kde.org)
5	* Copyright (C) 2003-2011, 2014 Apple Inc. All rights reserved.
6	*
7	* This library is free software; you can redistribute it and/or
8	* modify it under the terms of the GNU Library General Public
9	* License as published by the Free Software Foundation; either
10	* version 2 of the License, or (at your option) any later version.

```css
26  html {
27      display: block;
28  }
29
30  /* children of the <head> element all have display:none */
31  head, link, meta, script, style, title {
32      display: none;
33  }
34
35  /* generic block-level elements */
36
37  body {
38      display: block;
39      margin: 8px;
40  }
41
42  p {
43      display: block;
44      -webkit-margin-before: 1__qem;
45      -webkit-margin-after: 1__qem;
46      -webkit-margin-start: 0;
47      -webkit-margin-end: 0;
48  }
49
50  address, article, aside, div, footer, header, hgroup, layer, main, nav, section {
51      display: block;
52  }
53
54  marquee {
55      display: inline-block;
56      overflow: -webkit-marquee;
57  }
58
59  blockquote {
```

Equivalent to `margin-top: 1em`
& `margin-bottom: 1em`

http://www.iecss.com/opera-10.51.css

Go   MAR   OCT   OCT

20 captures
17 Oct 2011 – 22 Oct 2017

◀ 31 ▶

2015 2016 20 ▼ About this capture

```css
/* Copyright 2010 Opera Software */

a {
        color: #00C;
        text-decoration: underline;
}

a:visited {
        color: #800080;
}

abbr {
        border-bottom-style: dotted;
        border-bottom-width: 1px;
}

address {
        display: block;
        font-style: italic;
        padding-left: 2px;
        padding-right: 2px;
}

area {
        display: none;
}
```

http://www.iecss.com/opera-10.51.css

19 captures

17 Oct 2011 – 31 Oct 2016

MAR OCT NOV

31

2015 2016 20

About this capture

```
ot {
        padding: 0 40px 0 0;
}

optgroup {
        text-indent: 0;
}

option {
        text-indent: 0;
}

p {
        display: block;
        margin: 1em 0;
}
```

`1em` on top & bottom; `0` on right & left

```
pre {
        display: block;
        font-family: use-lang-def;
        font-size: 0.81em;
        margin: 1em 0;
        white-space: pre;
}

q {
```

HTML

Living Standard — Last Updated 24 December 2015

HTML
Living Standard — Last Updated 24 December 2015

?

"The CSS rules given in these subsections are, except where otherwise specified, expected to be used as part of the user-agent level style sheet defaults for all documents that contain HTML elements."

14.3.3 Flow content

```css
@namespace url(http://www.w3.org/1999/xhtml);

address, blockquote, center, div, figure, figcaption, footer, form, header, hr,
legend, listing, main, p, plaintext, pre, summary, xmp {
  display: block;
}

blockquote, figure, listing, p, plaintext, pre, xmp {
  margin-top: 1em; margin-bottom: 1em;
}

blockquote, figure { margin-left: 40px; margin-right: 40px; }

address { font-style: italic; }
listing, plaintext, pre, xmp {
  font-family: monospace; white-space: pre;
}

dialog:not([open]) { display: none; }
dialog {
  position: absolute;
  left: 0; right: 0;
  width: fit-content;
  height: fit-content;
  margin: auto;
  border: solid;
```

Users can specify CSS rules too

Why?

» All fonts are at least a certain size
» A certain font is used because it's more readable
» Always enable `text-decoration: underline` so links are obvious
» See outlines around elements with keyboard focus using `outline: solid`

Firefox has always allowed users to create their own styles in a file called `userContent.css` that goes into your Firefox Profile
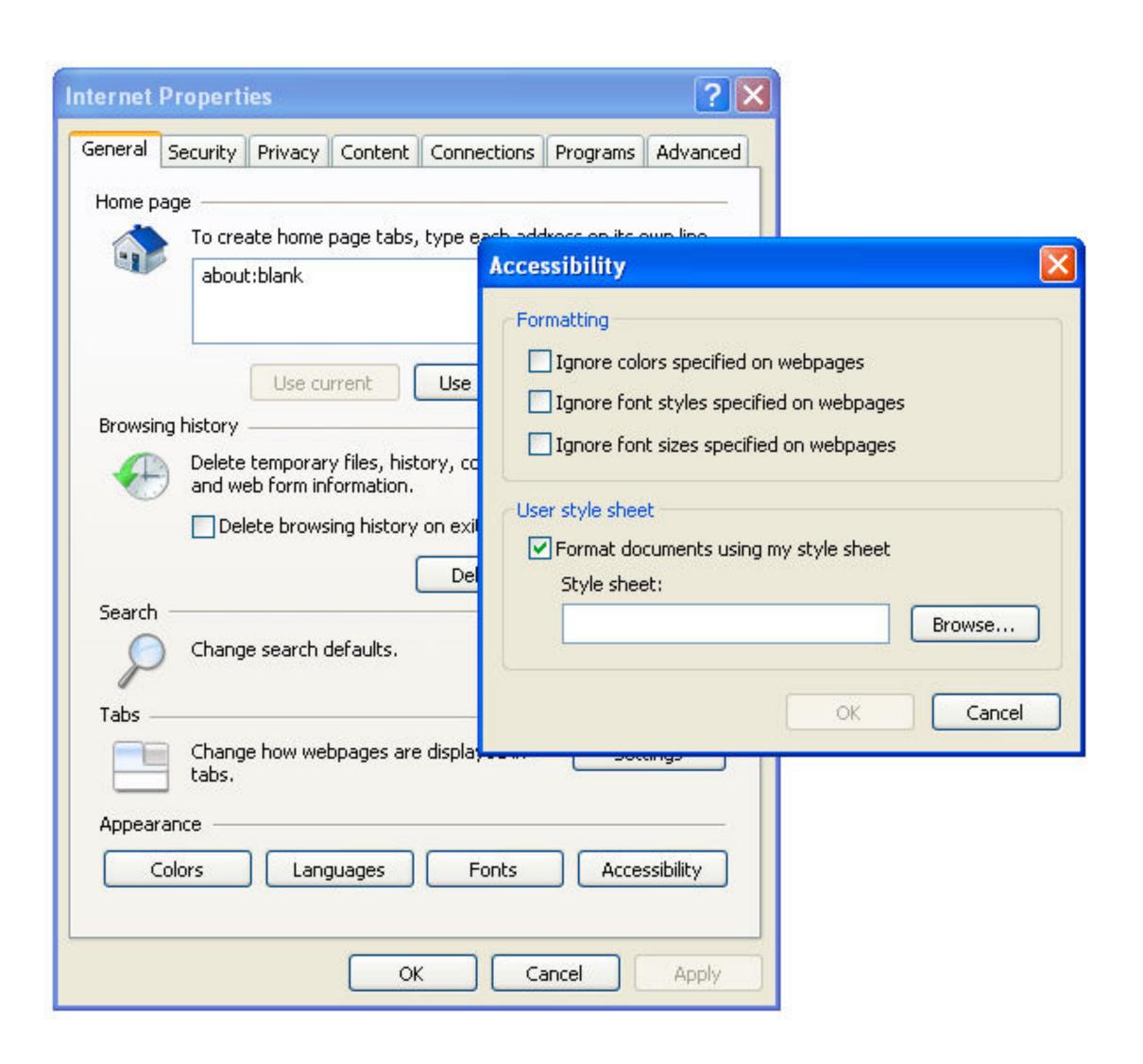
For more about your Profile: kb.mozillazine.org/Profile_folder_-_Firefox

For more about `userContent.css`: kb.mozillazine.org/UserContent.css

📝 SIDE NOTE

As of Firefox 69, `userContent.css` isn't supported by default unless users first enable the feature

1. Type `about:config` in the Firefox address bar & select Enter
2. Click the button that confirms you Accept the Risk and Continue
3. In the box at the top, search for `toolkit.legacyUserProfileCustomizations.stylesheets`
4. Double-click on the resulting line to toggle to `true`
5. Restart Firefox

Internet Explorer

Chrome 33 (2014) dropped support for a user styles file

Chrome 33 (2014) dropped support for a user styles file

Edge: *Never supported* a user styles file!

Edge: *Never supported* a user styles file!

Safari — I created `safari.css` for my own use

My `safari.css` file

```css
html {
  font-family: "Source Sans Pro", sans-serif;
}

h1, h2, h3, h4, h5, h6 {
  font-family: "Georgia Pro", serif;
}

code, kbd, pre, samp, tt, var {
  font-family: "Source Code Pro", monospace;
}
```

Browser makers' response when asked about missing support for a user styles file: "Get an extension"

Do get:

» Stylus for Chromium-based browsers & Firefox (which is excellent & removes all analytics, telemetry, & data collection)
» Cascadea for Safari

Do *not* get: ~~Stylish~~ (used to be good, but now it's spyware)

If a user's CSS contradicts a author's CSS, the user can make sure hers "wins"

Use `!important` after a property-value pair

```
p {
  font-size: 36px !important;
}
```

However, CSS authors can do the same thing!

My `safari.css` file

Notice: no `!important`

```css
html {
  font-family: "Source Sans Pro", sans-serif;
}

h1, h2, h3, h4, h5, h6 {
  font-family: "Georgia Pro", serif;
}

code, kbd, pre, samp, tt, var {
  font-family: "Source Code Pro", monospace;
}
```

The order of importance

AKA

The order in which stylesheets are *weighted*

Browser

↓

User

↓

Author

↓

Author `!important`

↓

User `!important`

Author & Author `!important`?

Why are you contradicting yourself?

Do I contradict myself?
Very well then I contradict myself,
(I am large, I contain multitudes.)

From Walt Whitman's "Song of Myself" from *Leaves of Grass*

You're not contradicting *yourself* (hopefully!)

What if you're using Bootstrap & its default CSS?

```
<link rel="stylesheet" href="bootstrap.css">
```

Is that CSS coming from the browser, user, or author?

You will want to override some of Bootstrap's selectors

```
<link rel="stylesheet" href="bootstrap.css">
<link rel="stylesheet" href="me.css">
```

Author vs. author

Sometimes, you will have to use `!important` to beat the *other* author

Be careful using `!important`

"`!important`: 3 seconds to type, 3 years to remove."
—Harry Roberts

"`!important` is the Hammer of Thor, smiting everything in its path." —Jans Carton

Only use it as a last resort

Specificity

Remember, compound & complex selectors are made up of multiple simple selectors

`p.foo.bar` (compound) is made up of 1 type & 2 classes

`.foo > cite` (complex) is composed of 1 class & 1 type

Specificity asks how *specific* is each selector?

Each simple selector is assigned a weight value

The greater the weight, the more specific the selector

In other words, with great weight comes great specificity 😜

Each ID selector is worth 100

Each class, pseudo-class, & attribute selector is worth 10

Each type & pseudo-element is worth 1

The following are *ignored* when calculating specificity:

» Combinators: `␣`, `>`, `+`, `~`
» Universal selector: `*`
» Negation pseudo-class (but not the contents — that is counted!): `:not(.but-this-part-is-counted)`

Inline styles (`<style="foo">`) always outweigh everything else (a good reason to hate them)

*		0
li	1	1
ul li	1+1	2
.foo > li	10+1	11
ul ol li.steps	1+1+1+10	13
li.steps.mech	1+10+10	21
#chapter1	100	100
style="foo"		∞

HTML

```html
<ul>
  <li>Great Old Ones
    <ul>
      <li>Abholos</li>
      <li>Alala</li>
      <li>Ammutseba</li>
      <li>Amon-Gorloth</li>
      <li>Aphoom-Zhah</li>
      <li>Apocolothoth</li>
      <li>Arwassa</li>
```

CSS

```css
/* Note: Using the universal selector
and direct child combinator may seem
more specific than the descendant
combinator. But since combinators do NOT
affect specificity, both of these
selectors have the same weight.
Therefore order determines which
selector wins. */

* ul > li { /* Specificity = 2 */
  color: blue;
}

ul li { /* Specificity = 2 */
  color: red;
}
```

JS

- **Great Old Ones**
  - Abholos
  - Alala
  - Ammutseba
  - Amon-Gorloth
  - Aphoom-Zhah
  - Apocolothoth
  - Arwassa
  - Atlach-Nacha
  - Ayi'ig
  - Aylith
  - Baoht Z'uqqa-Mogg
  - Basatan
  - B'gnu-Thun
  - Bokrug
  - Bugg-Shash
  - Byagoona
  - Byatis
  - Chaugnar Faugn
  - Coatlicue

Specificity Calculator

A visual way to understand _CSS specificity_. Change the selectors or paste in your own.

`nav > a:hover::before`

0 IDs

1 Classes, attributes and pseudo-classes

3 Elements and pseudo-elements

+ Duplicate

`ul#primary-nav li.active`

1 IDs

1 Classes, attributes and pseudo-classes

2 Elements and pseudo-elements

+ Duplicate

Specificity Calculator was built by Keegan Street. The specificity calculator JavaScript module is available on GitHub or via `npm install specificity`.

Order

Later CSS in the stylesheet wins over earlier CSS

```css
main.css:

.blue {color: blue}
.red {color: red}
```

```html
index.html:

<p class="red blue">
  What color am I?
</p>
```

```
main.css:

.blue {color: blue}
.red {color: red}
```

```
index.html:

<p class="red blue">
  What color am I?
</p>
```

```
main.css:

.red {color: red}
.blue {color: blue}
```

```
index.html:

<p class="blue red">
  What color am I?
</p>
```

```
main.css:

.red {color: red}
.blue {color: blue}
```

```
index.html:

<p class="blue red">
  What color am I?
</p>
```

The Cascade

The CSS Cascade

Sort by *Importance*

Browser
User
Author
Author !important
User !important

Match all CSS
declarations
for the DOM object

Sort by *Specificity*

*
li
ol li
.my_class
ol li.my_class
#my_id
style=""

Sort by *Order*

Rule #1
Rule #2
Rule #3
Rule #4

Display using
declaration
with most *weight*

winner

tie

winner

tie

winner

Here's some code I have on a website:

```html
<div class="callOut">
  <p>
      For the next 2 weeks...
  </p>
</div>
```

```css
.callOut {
  background-color: #E6E8F2;
  margin: 1em 1em 2em 1em;
  padding: 1em;
  border: 1px #ccc solid;
  border-radius: 1em;
}
```

Source: [xkcd](#) ⧉

> For the next 2 weeks, in addition to learning about wikis, you're also going to edit articles at Wikipedia. In order to fulfill this assignment, please [create an account at Wikipedia](#) ⧉ if you do not already have one. Do this as soon as possible, as you must have an account in existence for a few days before you can create/edit certain articles at Wikipedia.

Before you begin your reading, please read & comment: [Wiki & Wikipedia usage among students](#).

Definitions

Read all of these.

- [Computerworld Defines WikiWikiWeb](#)
  Wiki defined.
- [Wikis in Plain English](#)
  A short movie that explains the concept of wikis and how they work.

Why & how do wikis work?

Read all of these.

- [The Wiki Way](#)
  A review of a book about Wikis explains their philosophy and why they work.
- [Some useful & interesting wikis (besides Wikipedia)](#)
  A list of informational & useful wikis, besides Wikipedia.

The result!
However...

Source: xkcd

For the next 2 weeks, in addition to learning about wikis, you're also going to edit articles at Wikipedia. In order to fulfill this assignment, please create an account at Wikipedia if you do not already have one. Do this as soon as possible, as you must have an account in existence for a few days before you can create/edit certain articles at Wikipedia.

Before you begin your reading, please read & comment: Wiki & Wikipedia usage among students.

That extra space at the bottom really bothers me

Definitions

Read all of these.

- Computerworld Defines WikiWikiWeb
  Wiki defined.
- Wikis in Plain English
  A short movie that explains the concept of wikis and how they work.

Why & how do wikis work?

Read all of these.

- The Wiki Way
  A review of a book about Wikis explains their philosophy and why they work.
- Some useful & interesting wikis (besides Wikipedia)
  A list of informational & useful wikis, besides Wikipedia.

Source: xkcd

For the next 2 weeks, in addition to learning about wikis, you're also going to edit articles at Wikipedia. In order to fulfill this assignment, please create an account at Wikipedia if you do not already have one. Do this as soon as possible, as you must have an account in existence for a few days before you can create/edit certain articles at Wikipedia.

46

For the next 2 weeks, in addition to learning about wikis, you're also going to edit articles at Wikipedia. In order to fulfill this assignment, please create an account at Wikipedia if you do not already have one. Do this as soon as possible, as you must have an account in existence for a few days before you can create/edit certain articles at Wikipedia.

61

Why & how do wikis work?

Read all of these.

- The Wiki Way
  A review of a book about Wikis explains their philosophy and why they work.
- Some useful & interesting wikis (besides Wikipedia)
  A list of informational & useful wikis, besides Wikipedia.

To fix it, I put this in my CSS at lines 194–196:

```
.callOut > p:last-child {
  margin-bottom: 0;
}
```

Let me explain what that means…

`>` is a *combinator* that selects the direct children of (*not* descendants of) the `.callOut` class

`p:last-child` is a *pseudo-class* that means the `<p>` that is the last child of (*not* descendant of) the `.callOut` class

```css
.callOut > p:last-child {
  margin-bottom: 0;
}
```

So this selects the `<p>` that is the *last direct child* of the `.callOut` class

```css
.callOut > p:last-child {
  margin-bottom: 0;
}
```

```html
<div class="callOut">
  <p>foo</p>
  <p>bar</p>
  <p>baz</p>
  <p>quz</p>          ⬅ Selected!
  <blockquote>
    <p>quux</p>       ⬅ Not selected, as this <p> is
  </blockquote>            *not* the last *direct child*
</div>
```

Why use this?

```
.callOut > p:last-child {
  margin-bottom: 0;
}
```

Why not just use this?

```
.callOut > p {
  margin-bottom: 0;
}
```

Because sometimes there are 2 or more paragraphs inside
`.callOut`

So back to where I was… I put this in my CSS at line 194:

```
.callOut > p:last-child {
  margin-bottom: 0;
}
```

Let's check the webpage…

For the next 2 weeks, in addition to learning about wikis, you're also going to edit articles at Wikipedia. In order to fulfill this assignment, please create an account at Wikipedia if you do not already have one. Do this as soon as possible, as you must have an account in existence for a few days before you can create/edit certain articles at Wikipedia.

Nothing changed!

Why didn't it work?

Let's open the Inspector & find out why

Source: xkcd

For the next 2 weeks, in addition to learning about wikis, you're also going to edit articles at Wikipedia. In order to fulfill this assignment, please create an account at Wikipedia if you do not already have one. Do this as soon as possible, as you must have an account in existence for a few days before you can create/edit certain articles at Wikipedia.

p 574px × 120px

Before you begin your reading, please read & comment: Wiki & Wikipedia usage among students.

Definitions

- Broadcast Flag
- Bullying
- Business-Education Partnerships
- Business Intelligence
- Business School
- Cameras
- Capitalism
- Cartels
- CBDTPA
- Cell Phones
- Climate Change
- Cloud Computing
- Collaboration
- Comic Books
- Communications
- Concepts
- Consolidation
- Content Management Systems

Elements   Network   Resources   Timelines   Debugger   Storage   Console

div#post-10096.post-10096.post.type-post.status-publish... > div.entry-content > div.callOut > p

```
document.getElementById('primary').style.display = 'block'; }else{ document.getElementById('primary').style.display =
'none'; document.getElementById('content').style.margin = '0px 200px';}">Toggle Visibility</a> -->
    <div class="entry-meta">…</div>
    <!-- .entry-meta -->
    <div class="entry-content">
        <p>…</p>
        <div class="callOut">
            <p>…</p> = $0
        </div>
        <p>…</p>
        <h2>Definitions</h2>
        <p>Read all of these.</p>
        <ul>…</ul>
        <h2>Why & how do wikis work?</h2>
        <p>Read all of these.</p>
```

Node   Styles   Layers

Computed   Rules   Metrics

#content p — style.css:146
margin-bottom: 12px;

.callOut > p:last-child — style.css:194
margin-bottom: 0;

Filter Styles

Main Frame

```css
{} #content p — style.css:146
    margin-bottom: 12px;
```

```css
{} .callOut > p:last-child — style.css:194
    margin-bottom: 0;
```

Hmmm... mine is being beaten by earlier code

Lines 146–148:
```css
#content p {
  margin-bottom: 12px;
}
```

Lines 194–196:
```css
.callOut > p:last-child {
  margin-bottom: 0;
}
```

Why isn't mine winning since it's later in order?

Lines 146–148:
```css
#content p {
  margin-bottom: 12px;
}
```

Lines 194–196:
```css
.callOut > p:last-child {
  margin-bottom: 0;
}
```

Is *importance* causing the 1st declaration block to win?

Nope

Lines 146–148:
```
#content p {
  margin-bottom: 12px;
}
```

Lines 194–196:
```
.callOut > p:last-child {
  margin-bottom: 0;
}
```

Is *specificity* causing the 1st declaration block to win?

Let's look

```
#content p {
  margin-bottom: 12px;
} ;
```

```
.callOut > p:last-
child {
    margin-bottom: 0;
}
```

Specificity

(#content = 100) +
(p = 1) =
101

Specificity

(.callout = 10) +
(p = 1) +
(:last-child = 10) =
21

Lines 146–148:

```
#content p {
  margin-bottom: 12px;
}
```

Lines 194–196:

```
.callOut > p:last-child {
  margin-bottom: 0;
}
```

Because *specificity* meant the 1st declaration block had the most weight, *order* never entered into the picture

The CSS Cascade

Sort by *Importance*

Browser
User
Author
Author !important
User !important

Nope—tied

Match all CSS
declarations
for the DOM object

Sort by *Specificity*

*
li
ol li
.my_class
ol li.my_class
#my_id
style=""

tie

winner

winner

Display using
declaration
with most *weight*

Specificity beat me

tie

Sort by *Order*

Rule #1
Rule #2
Rule #3
Rule #4

winner

Never got to it

There are 3 ways to solve this problem

Lines 146–148:
```
#content p {
  margin-bottom: 12px;
}
```

Lines 194–196:
```
.callOut > p:last-child {
  margin-bottom: 0 !important;
}
```

Adding `!important` makes the 2nd ruleset win due to *importance*, so *specificity & order* never come into play

The CSS Cascade

Sort by *Importance*

Browser
User
Author
Author !important
User !important

I won before
we could get to
anything else

**Match all CSS
declarations
for the DOM object**

tie

Sort by *Specificity*

*
li
ol li
.my_class
ol li.my_class
#my_id
style=""

winner

**Display using
declaration
with most *weight***

winner

tie

Never got to it

Sort by *Order*

Rule #1
Rule #2
Rule #3
Rule #4

winner

Lines 146–148:

```
#content p {
  margin-bottom: 12px;
}
```

Lines 194–196:

```
#content .callOut > p:last-child {
  margin-bottom: 0;
}
```

Adding `#content` to line 194 makes the 2nd ruleset win due to *specificity*, so *order* never comes into play

Lines 146–148:

```
#content p {
  margin-bottom: 12px;
}
```

100 + 1 = 101

100 + 10 + 1 + 10 = 121

Lines 194–196:

```
#content .callOut > p:last-child {
  margin-bottom: 0;
}
```

Adding `#content` to line 194 makes the 2nd ruleset win due to *specificity*, so *order* never comes into play

The CSS Cascade

Sort by *Importance*

Browser
User
Author
Author !important
User !important

← Tied

Match all CSS declarations for the DOM object

Sort by *Specificity*

*
li
ol li
.my_class
ol li.my_class
#my_id
style=""

I won (121 over 101)

Sort by *Order*

Rule #1
Rule #2
Rule #3
Rule #4

Never got to it →

Display using declaration with most *weight*

tie

tie

winner

winner

winner

Lines 146–148:

```css
p {
  margin-bottom: 12px;
}
```

Lines 194–196:

```css
.callOut > p:last-child {
  margin-bottom: 0;
}
```

Getting rid of `#content` on line 146 makes the 2nd ruleset win due to *specificity*, but I can't easily remove it!

Lines 146–148:

```
p {                              1
  margin-bottom: 12px;
}
```

Lines 194–196:   10 + 1 + 10 = 21

```
.callOut > p:last-child {
  margin-bottom: 0;
}
```

Getting rid of #content on line 146 makes the 2nd ruleset win due to *specificity*, but I can't easily remove it!

Source: xkcd ⧉

For the next 2 weeks, in addition to learning about wikis, you're also going to edit articles at Wikipedia. In order to fulfill this assignment, please create an account at Wikipedia ⧉ if you do not already have one. Do this as soon as possible, as you must have an account in existence for a few days before you can create/edit certain articles at Wikipedia.

Much better!

Before you begin your reading, please read & comment: Wiki & Wikipedia usage among students.

Definitions

Read all of these.

- Computerworld Defines WikiWikiWeb
  Wiki defined.
- Wikis in Plain English
  A short movie that explains the concept of wikis and how they work.

Why & how do wikis work?

Read all of these.

- The Wiki Way
  A review of a book about Wikis explains their philosophy and why they work.
- Some useful & interesting wikis (besides Wikipedia)
  A list of informational & useful wikis, besides Wikipedia.

Source: xkcd ⧉

For the next 2 weeks, in addition to learning about wikis, you're also going to edit articles at Wikipedia. In order to fulfill this assignment, please create an account at Wikipedia ⧉ if you do not already have one. Do this as soon as possible, as you must have an account in existence for a few days before you can create/edit certain articles at Wikipedia.

46

For the next 2 weeks, in addition to learning about wikis, you're also going to edit articles at Wikipedia. In order to fulfill this assignment, please create an account at Wikipedia ⧉ if you do not already have one. Do this as soon as possible, as you must have an account in existence for a few days before you can create/edit certain articles at Wikipedia.

37

Why & how do wikis work?

Read all of these.

- The Wiki Way
  A review of a book about Wikis explains their philosophy and why they work.
- Some useful & interesting wikis (besides Wikipedia)
  A list of informational & useful wikis, besides Wikipedia.

Bonus question: how do we get rid of those extra pixels at the top?

Source: xkcd

For the next 2 weeks, in addition to learning about wikis, you're also going to edit articles at Wikipedia. In order to fulfill this assignment, please create an account at Wikipedia if you do not already have one. Do this as soon as possible, as you must have an account in existence for a few days before you can create/edit certain articles at Wikipedia.

46

For the next 2 weeks, in addition to learning about wikis, you're also going to edit articles at Wikipedia. In order to fulfill this assignment, please create an account at Wikipedia if you do not already have one. Do this as soon as possible, as you must have an account in existence for a few days before you can create/edit certain articles at Wikipedia.

37

Why & how do wikis work?

Read all of these.

- The Wiki Way
  A review of a book about Wikis explains their philosophy and why they work.
- Some useful & interesting wikis (besides Wikipedia)
  A list of informational & useful wikis, besides Wikipedia.

Then I realized that `<div class="callOut">` doesn't always end with `<p>`

```css
#content .callOut > *:last-child {
  margin-bottom: 0;
}
```

Now it's weighted to win *and* selecting the correct DOM objects

What if I have nested last children inside other nested last children?

```css
.callOut > *:last-child,
.callOut > *:last-child > *:last-child,
.callOut > *:last-child > *:last-child >
*:last-child {
  margin-bottom: 0;
}
```

CSS Reset Keywords

CSS keywords have special meaning in CSS & are reserved by the language, e.g.:

» `border` & `font-family` are *property* keywords
» `solid` & `dotted` are *value* keywords
» `black` & `red` are *color value* keywords
» `<length>` & `<color>` are *data type* keywords
» `@media` & `@supports` are *at-rule* keywords
» `attr()` & `calc()` are *function* keywords
» `:hover` & `:focus` are *pseudo-class* keywords
» `::before` & `::first-letter` are *pseudo-element* keywords

There are 5 global keywords in CSS

» 4 are values: `initial`, `inherit`, `unset`, & `revert`
» 1 is a property: `all`

`initial` value resets property back to its default value as defined in the spec

`inherit` value forces an element to inherit styles from an ancestor that it would not normally inherit

`unset` value acts as either `inherit` (if the property is inherited) or `initial` (if the property is not)

`revert` value rolls back the property's cascade, depending upon who declared it: the browser, user, or author

`initial` value resets property *back to its default value as defined in the spec*

Remember, in the specs for CSS...

» `display` default is `inline`
» `font-size` default is `medium`
» `border-style` default is `none`
» `border-width` default is `medium`
» `margin` & `padding` defaults are `0`
» `font-family` default "depends on user agent"

caption-side

English ▼

On this Page

Syntax

Formal definition

Formal syntax

Examples

Specifications

Browser compatibility

The `caption-side` CSS property puts the content of a table's `<caption>` on the specified side. The values are relative to the `writing-mode` of the table.

CSS Demo: caption-side

Reset

```
caption-side: top;
```

```
caption-side: bottom;
```

Famous animals

Name	Location
Giraffe	Africa
Penguin	Antarctica
Sloth	South America
Tiger	Asia

Related Topics

CSS

CSS Reference

CSS Table

Formal definition

Initial value	top
Applies to	table-caption elements
Inherited	yes
Computed value	as specified
Animation type	discrete

Formal syntax

```
top | bottom | block-start | block-end | inline-start | inline-end
```

Don't get confused

Look at `<p>`

» Browsers set defaults by *selector*, so `<p>` defaults to
`display: block`
» The spec sets defaults by *property*, so `display:`
`inline` is the default

`initial` resets properties back to the spec, not the
browser!

HTML

```html
<p>
  Now that I am telling it anyway, lest the brooding
make me a maniac, I wish I had never concealed it.
</p>
<p>
  For I, and I only, know what manner of fear lurked on
that spectral and desolate mountain.
</p>
```

CSS (SCSS) `Compiled`

```scss
html {
  font-size: 24px;
  font-family: "Iowan Old Style", Georgia, serif;
}

p {
  font-size: initial;
  font-family: initial;
  display: initial;
}
```

JS

Now that I am telling it anyway, lest the brooding make me a maniac, I wish I had never concealed it. For I, and I only, know what manner of fear lurked on that spectral and desolate mountain.

The browser's default for `<p>` is `display: block`

The spec says that all boxes default to `display: inline`

`initial` resets back to the *spec*, not the browser

The `inherit` value forces an element to inherit values from properties that are not normally inherited

HTML

```html
<blockquote>
  <p>
    Of the five remaining pictures, three were of swamp
and hill scenes which seemed to bear traces of hidden
and <em>unwholesome</em> tenancy. Another was of a queer
mark in the ground very near Akeley's house…
  </p>
</blockquote>
```

CSS

```css
html {
  font-size: 24px; /* inherited */
}

blockquote {
  font-family: "Iowan Old Style", Georgia, serif; /* inherited */
  border: 1px solid black; /* not inherited */
  padding: 0 1em; /* not inherited */
}

em, p {
  border: inherit; /* p inherits from blockquote; em inherits from p */
}
```

JS

Of the five remaining pictures, three were of swamp and hill scenes which seemed to bear traces of hidden and *unwholesome* tenancy. Another was of a queer mark in the ground very near Akeley's house…

`unset` value acts as either `inherit` (if the property is inherited) or `initial` (if the property is not)

What's the use case for this? See `all` just ahead

`revert` value rolls back the property's cascade, depending upon who declared it: the browser, user, or author

» if the author is the origin, it rolls back to the user
» if the user is the origin, it rolls back to the browser
» if the browser is the origin, it rolls back to the spec

If the origin that is rolled back to does not declare a value for the property, it is skipped & the rendering engine rolls back another level

Effects of `revert`

Author

User

Browser

Spec

If `revert` is rolling back a property's style set by an author, & the user has not set a style, it skips the user & rolls back to the browser, & so on

Most common use-case for `revert`: you have a heavily-modified selector & you want to revert back to the browser's defaults (remember, most users never set any styles, so that one is skipped over)

`all`

Magic property that resets all property values, except for direction & unicode-bidi

Really designed to be used with small components — you wouldn't want to do this for an entire page

Values are the 4 global keywords: `initial`, `inherit`, `unset` (which now makes sense in a limited scenario!), & `revert`

HTML

```
1  <nav>
2    <menu>
3      <li><a href="❌">Search</a></li>
4      <li><a href="❌">Log In</a></li>
5      <li><a href="❌">Shopping Cart</a></li>
6    </menu>
7  </nav>
8
9  <nav class="nav-widget">
10   <menu>
11     <li><a href="❌">Search</a></li>
12     <li><a href="❌">Log In</a></li>
13     <li><a href="❌">Shopping Cart</a></li>
14   </menu>
15 </nav>
```

CSS (SCSS)

```
1  html {
2    /* font-size & font-family are inherited */
3    font-size: 24px;
4    font-family: "Iowan Old Style", Georgia, serif;
5  }
6
7
```

JS

- Search
- Log In
- Shopping Cart

- Search
- Log In
- Shopping Cart

```
6      </menu>
7    </nav>
8
9   <nav class="nav-widget">
10     <menu>
11       <li><a href="❌">Search</a></li>
12       <li><a href="❌">Log In</a></li>
13       <li><a href="❌">Shopping Cart</a></li>
14     </menu>
15   </nav>
```

CSS (SCSS) `Compiled`

```scss
1  html {
2    /* font-size & font-family are inherited */
3    font-size: 24px;
4    font-family: "Iowan Old Style", Georgia, serif;
5  }
6
7  .nav-widget * {
8    /* Resets all uninherited properties below back to spec */
9    all: unset;
10 }
11
```

JS

- Search
- Log In
- Shopping Cart

Search Log In Shopping Cart

	IE	Edge	Firefox	Safari	iOS	Chrome	Android
initial	–	12	19	3.2	4	4	2.3
inherit	8	12	2	3.1	3.2	4	2.1
unset	–	13	27	9.1*	9.3*	41	41
revert	–	–	67	9.1	9.3	84?	–
all	–	79	27	9.1	9.3	37	4.4.4

* `all: unset` is buggy in Safari & sets `color` to `black`, preventing you from setting another `color`; the workaround is to use `-webkit-text-fill-color` everyplace you also use `color`; Safari 14 fixes this

Tools

Books

Great overview of HTML5 & CSS2 (& some CSS3)

References

MDN MOZILLA DEVELOPER NETWORK

ZONES   WEB PLATFORM   DEVELOPER PROGRAM   TOOLS   DEMOS   🔍

developer.mozilla.org/en-US/docs/Web/CSS/Reference

CSS reference

As of July 2020 there are 617 terms!

This *CSS Reference* lists all standard CSS properties, pseudo-classes and pseudo-elements, @-rules, units, and selectors in alphabetic order. It allows you to quickly access detailed information for each of them. It not only lists the CSS 1 and CSS 2.1 properties, but also is a CSS3 reference that links to any CSS3 property and concept standardized, or already stabilized.

See also Mozilla CSS Extensions for Gecko-specific properties prefixed with `-moz`; and WebKit CSS Extensions for WebKit-specific properties. See Vendor-prefixed CSS Property Overview by Peter Beverloo for all prefixed properties.

A	counter-reset	:last-child	repeating-radial-gradient()
:active	cross-fade()	:last-of-type	:required
::after (:after)	cubic-bezier()	left	resize
align-content	cursor	:left	<resolution>
align-items	cycle()	<length>	rgb()
align-self		letter-spacing	rgba()
all	D	linear-gradient()	right
<angle>	:default	:link	:right
	deg		

mozilla W3C® Microsoft

SAMSUNG Google

MDN supporters & contributors

Fresh Article July 6, 2020

Bootstrap 5

Author	Comments
Chris Coyier	Start Conversation ·→

It's always notable when the world biggest CSS framework goes up a major version (it's in alpha now).

It has dropped jQuery and IE, started using some CSS custom properties, gone fully customized with form elements, started to embrace utility classes, and includes a massive icon set you can use via SVG sprite. Sweet....

Read article

BOOTSTRAP

Article Jun 26, 2020

In Defense of a Fussy Website

Author
Sarah Drasner

CARE

INTENTION

PORTFOLIO

UX

Article Jun 17, 2020

Advice for Complex CSS Illustrations

Author
Jhey Tompkins

ART

CSS SHAPES

Article Jul 1, 2020

A Complete Guide to Dark Mode on the Web

Author Adhuham

DARK MODE

Article Jun 15, 2020

What is Developer Experience (DX)?

Author
Chris Coyier

DX   UX

Article Jun 12, 2020

Striking a Balance Between Native and Custom Select Elements

Author
Sandrina Pereira

ACCESSIBILITY

FORMS

SELECT

1st 8 Guides on CSS Tricks (there are lots more!):

» A Complete Guide to Flexbox ★
» Media Queries for Standard Devices
» A Complete Guide to Grid ★
» A Complete Guide to the Table Element
» Centering in CSS: A Complete Guide
» A Complete Guide to SVG Fallbacks
» A Nerd's Guide to Color on the Web
» A Complete Guide to Data Attributes

Can I use

? ⚙ Settings

CSS

- ::placeholder CSS pseudo-element
- ::selection CSS pseudo-element
- :placeholder-shown CSS pseudo-class
- @font-face Web fonts
- Blending of HTML/SVG elements
- calc() as CSS unit value
- 2.1 selectors
- background-attachment
- background-blend-mode
- background-position edge offsets
- box-decoration-break
- clip-path property
- Counter Styles
- Counters
- currentColor value

HTML5

- Audio element
- Audio Tracks
- Autofocus attribute
- Canvas (basic support)
- Canvas blend modes
- classList (DOMTokenList)
- Color input type
- contenteditable attribute (basic support)
- Custom Elements
- Custom protocol handling
- Datalist element
- dataset & data-* attributes
- Date and time input types
- Details & Summary elements
- Dialog element

SVG

- Inline SVG in HTML5
- SVG (basic support)
- SVG effects for HTML
- SVG favicons
- SVG filters
- SVG fonts
- SVG fragment identifiers
- SVG in CSS backgrounds
- SVG in HTML img element
- SVG SMIL animation
- **All SVG features**

JS API

- Ambient Light API

Can I use   grid   ?   ⚙ Settings

2 results found

CSS Grid Layout 📄 - CR

Global   69.53% + 5.62% = 75.15%

unprefixed: 69.53%

Method of using a grid concept to lay out content, providing a mechanism for authors to divide available space for layout into columns and rows using a set of predictable sizing behaviors

Current aligned | Usage relative | Date relative | **Show all**

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
			[1] 49 🚩			10.2			
	[2] 15 ▬	55	60	10.1	47	10.3		4.4	
[2] 11 ▬	16	56	61	11	48	11	all	56	61
		57	62	TP	49				
		58	63		50				
		59	64						

Notes | Known issues (2) | Resources (12) | Feedback

```css
/* Styles */

@import "extra.css";

ul li
{
    list-style: square;
    margin: 2em 20% 15px 0;
}

#content
{
    -webkit-font-smoothing: antialiased;
    background: url(img/gradient.png);
    background: linear-gradient(to bottom, red, rgba(255,0,0,0));
}


@media only screen and (min-width: 35em)
{
    #content
    {
        width: 50%;
    }
}
```

CSS VOCABULARY

?

Comment

Statement

Rule-set

At-rule

At-keyword

Media query

Media query list

Media type

Expression

Media feature

Block

Declaration block

Selector

Simple selector

Type selector

Universal selector

apps.workflower.fi/vocabs/css/en

```css
/* Styles */

@import "extra.css";

ul li
{
    list-style: square;
    margin: 2em 20% 15px 0;
}

#content
{
    -webkit-font-smoothing: antialiased;
    background: url(img/gradient.png);
    background: linear-gradient(to bottom, red, rgba(255,0,0,0));
}

@media only screen and (min-width: 35em)
{
    #content
    {
        width: 50%;
    }
}

.item + .item,
.item ~ .item
```

CSS VOCABULARY

?

Comment

Statement

Rule-set

At-rule

At-keyword

Media query

Media query list

Media type

Expression

Media feature

Block

Declaration block

Selector

Simple selector

Type selector

Universal selector

ID selector

Class selector

Attribute selector

```css
/* Styles */

@import "extra.css";

ul li
{
    list-style: square;
    margin: 2em 20% 15px 0;
}

#content
{
    -webkit-font-smoothing: antialiased;
    background: url(img/gradient.png);
    background: linear-gradient(to bottom, red, rgba(255,0,0,0));
}


@media only screen and (min-width: 35em)
{
    #content
    {
        width: 50%;
    }
}

.item + .item,
.item ~ .item
```

Class selector

Attribute selector

Pseudo-class

Pseudo-element

Combinator

Descendant combinator

Child combinator

Adjacent sibling combinator

General sibling combinator

Declaration

Property

Value

Function

Keyword

String

URL

Number

Percentage

Length

Unit

Color

Vendor prefix

rsgranne + css 192

RSS

« earlier

About CodePen

CodePen is a playground for the front end side of the web. It's all about inspiration, education, and sharing.

Need to build a reduced test case to demonstrate and figure out a bug? CodePen is great for that. Want to show off your latest creation and get feedback from your peers? CodePen is great for that. Want to find example of a particular design pattern for you project? CodePen is great for that.

CodePen PRO takes things further with features like Collab Mode which allows you to pair program in real time and Professor Mode for allowing a group of students to follow you as you teach code and chat with each other.

code css design javascript webdev html collaboration service websanity teaching webster

24 days ago by rsgranne copy to mine

Download Google Fonts - Fonts.com

Sync Google Fonts to your Desktop with SkyFonts

We are proud to have teamed up with Google to offer desktop versions of their popular Google Fonts free of charge. Offered for use in print, these fonts are delivered using SkyFont's patent-pending font delivery technology and can be used anywhere.

Each time a font is updated — such as when new characters are added — SkyFonts will automatically update the font on your device. Syncing Google Fonts with SkyFonts will also improve your web browsing experience, by cutting the time spent downloading fonts.

fonts google typography css

24 days ago by rsgranne copy to mine

BuiltWith Web Technology Usage Statistics

Search All

related tags

addons ⊕ adobe ⊕ advertising ⊕ ajax ⊕ animation

⊕ api ⊕ apple ⊕ art ⊕ assignment ⊕ blog

bookmarks ⊕ books ⊕ browser ⊕ business

characterentities ⊕ chart ⊕ cheatsheet

chrome ⊕ cms ⊕ code ⊕ collaboration

collection ⊕ color ⊕ commercial ⊕ compression

⊕ conversion ⊕ creativity ⊕ CSS ⊖ demo

design ⊕ dhtml ⊕ diagnostic ⊕ dictionary

directory ⊕ diy ⊕ documentation ⊕ download

⊕ ecommerce ⊕ editing ⊕ education ⊕ email

ericmeyer ⊕ examples ⊕ extensions

pinboard.in/u:rsgranne/t:css

Browser Tools

Built-in developer tools are excellent

Home :: Scott Granneman   +

www.granneman.com

Google ▾ Google

p ▾

Copy Inner HTML
Copy Outer HTML
Copy Unique Selector

Delete Node

:hover
:active
:focus

Scott Granneman is an author, educator, & partner in a Web development company in St. Louis, Missouri. Scott has written six books, contributed to two others, and writes columns for SecurityFocus and *Linux Magazine*. He is an Adjunct Professor at Washington University in St. Louis & Webster University, teaching courses on technology, security, web development, & the Internet. As a Principal of WebSanity, he plans, develops, and hosts websites for clients in 12 states using an open source, UNIX-based content management system. He has given presentations all over the United States, and loves speaking to people.

News & Updates

Note: The following news items are personal in nature & are automatically selected from GranneBlog.

Subscribe to News via:

Key Site Links

Pages you're probably looking for:

- Publications
- Summer Vacation 2007: The Plains
- Web Development Key Links
- Wash U: Website Design & Development Syllabus
- Webster U: Programming for Web Communications Syllabus

✕ ▢ ▤ ⚙   › Console   ▭ Inspector   ⏸ Debugger   ✎ Style Editor   ⊙ Profiler   ≡ Network

‹ ...ayout-row.ccm-layout-row-1   div.ccm-layout-5-col-1.ccm-layout-cell   div.ccm-layout-col-spacing   div   p ›

Rules   Computed   Fonts   Box Model

```
<div id="ccm-layout-body-5-2" class="ccm-layout ccm-layout-table ccm-layout-name-Body-Layout-2 ">
  <div class="ccm-layout-row ccm-layout-row-1">
    <div class="ccm-layout-5-col-1 ccm-layout-cell ccm-layout-col ccm-layout-col-1 first" style="width:71%">
      <div class="ccm-layout-col-spacing">
        <div>
          <p></p>
        </div>
        <div></div>
        <div id="rssSummaryList1170" class="rssSummaryList"></div>
      </div>
    </div>
```

```
element {
}

p, dl, li, th, td,          typography.css:114
blockquote {
  ▸ font: 16px/1.5em "Source Sans
Pro","Lucida Grande",Lucida,Verdana,sans-
serif;
  color: rgb(51, 51, 51);
}

p, blockquote, ul, ol,      typography.css:109
dl, fieldset, address,
pre {
  margin-bottom: 15px;
  color: rgb(51, 51, 51);
```

www.granneman.com

Bonjour | S-Foc | Ent | RSG | G | WS | Soc | Apps | lets | News | Save | Open URL

Scott Grannen

Writing | Presentations | Teaching | Web Development | Tech Info | Personal

Scott Granneman is an author, educator, & partner in a Web development company in St. Louis, Missouri. Scott has written six books, contributed to two, and penned monthly columns for SecurityFocus and *Linux Magazine*. He is an Adjunct Professor at Washington University in St. Louis & Webster University, teaching courses on technology, security, web development, & the Internet. As a Principal of WebSanity, he plans, develops, and hosts websites for clients in 12 states using an open source, UNIX-based content management system. He has given presentations all over the United States, and loves speaking to people.

Key Site Links

Pages you're probably looking for:

- Publications
- Summer Vacation 2007: The Plains
- Web Development Key Links

18   1   0

Computed | **Rules** | Metrics

www.granneman.com

Images
Scripts
Stylesheets
Cookies — www.granneman.com
Extension Scripts
Local Storage — www.granneman.com
Session Storage — www.granneman.com

```
<script>…</script>
<head>…</head>
<body id="toc-top" class="cID-1">
  <div id="not_footer">
    <div id="header">…</div>
    <div id="site_nav">…</div>
    <div id="page">
      <div id="page_content" class="home_page">
        <div id="body_copy">
          <div id="ccm-layout-wrapper-14" class="ccm-layout-wrapper">
            <div id="ccm-layout-body-5-2" class="ccm-layout ccm-layout-table ccm-layout-name-Body-Layout-2 ">
              <div class="ccm-layout-row ccm-layout-row-1">
                <div class="ccm-layout-5-col-1 ccm-layout-cell ccm-layout-col ccm-layout-col-1 first" style="width:71%">
```

p — Style Attribute
No Properties — Click to Edit

+ New Rule

p, dl, li, th, td, blockquote — typography.css:114
font: 16px/1.5em "Source Sans Pro", "Lucida Grande", Lucida, Verdana, sans-serif;
color: #333;

p, blockquote, ul, ol, dl, fieldset, address, pre — typography.css:109

Filter Resource List

Home :: Scott Granneman

www.granneman.com

Apps   S-Foc   Ent   RSG   G   WS   Soc   Apps   lets   News   Save   Open URL

Scott Granneman

p 691px × 168px

Scott Granneman is an author, educator, & partner in a Web development company in St. Louis, Missouri. Scott has written six books, contributed to two, and penned monthly columns for SecurityFocus and *Linux Magazine*. He is an Adjunct Professor at Washington University in St. Louis & Webster Univers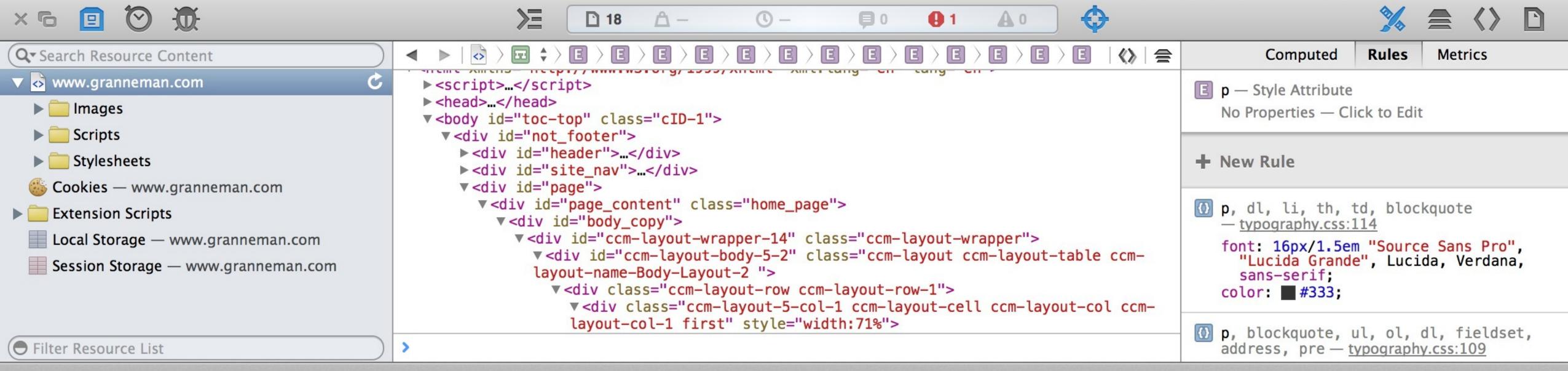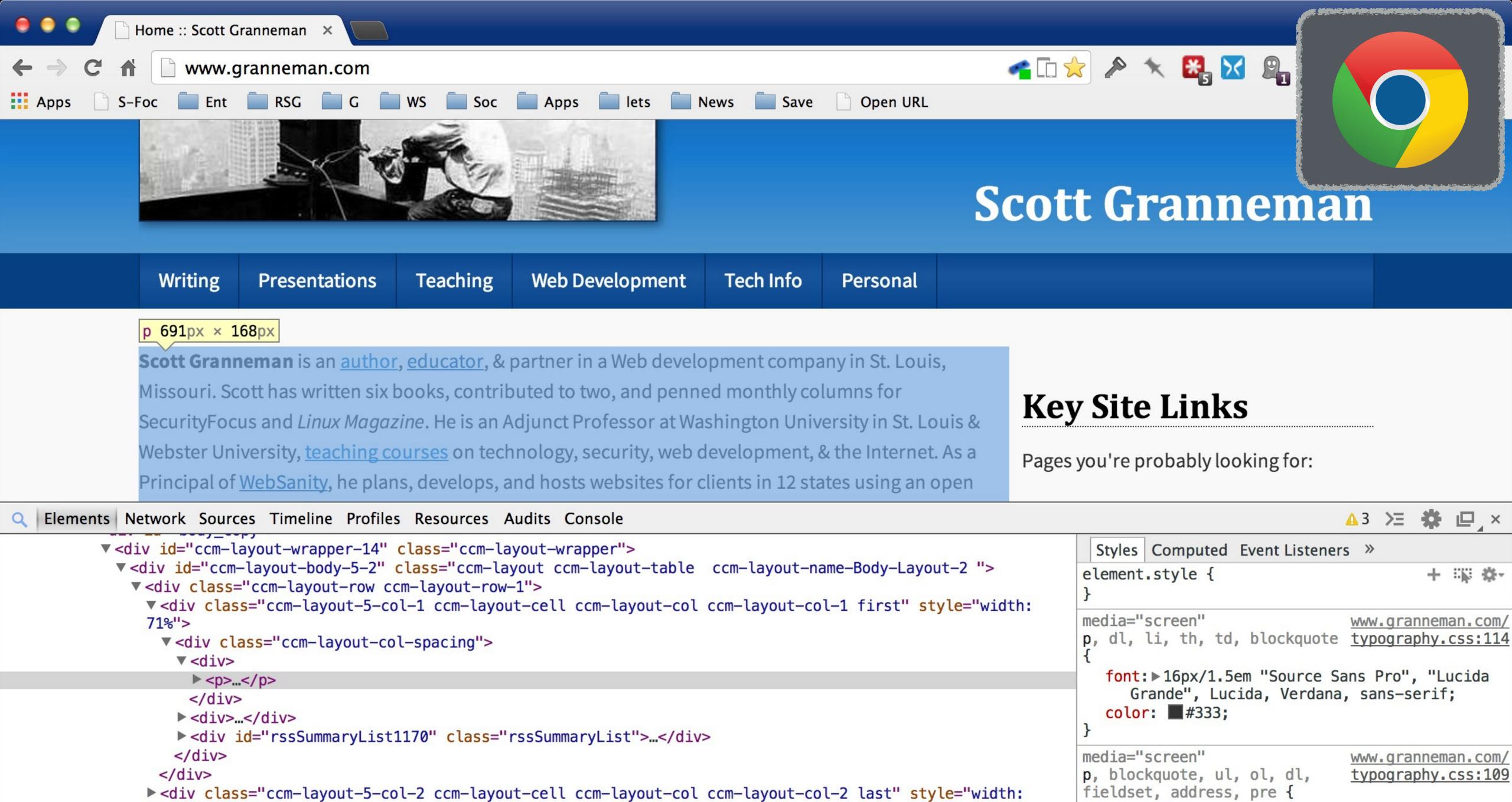ity, teaching courses on technology, security, web development, & the Internet. As a Principal of WebSanity, he plans, develops, and hosts websites for clients in 12 states using an open

Writing   Presentations   Teaching   Web Development   Tech Info   Personal

Key Site Links

Pages you're probably looking for:

Elements   Network   Sources   Timeline   Profiles   Resources   Audits   Console   ⚠3

```
<div id="ccm-layout-wrapper-14" class="ccm-layout-wrapper">
  <div id="ccm-layout-body-5-2" class="ccm-layout ccm-layout-table  ccm-layout-name-Body-Layout-2 ">
    <div class="ccm-layout-row ccm-layout-row-1">
      <div class="ccm-layout-5-col-1 ccm-layout-cell ccm-layout-col ccm-layout-col-1 first" style="width: 71%">
        <div class="ccm-layout-col-spacing">
          <div>
            <p>…</p>
          </div>
          <div>…</div>
          <div id="rssSummaryList1170" class="rssSummaryList">…</div>
        </div>
      </div>
      <div class="ccm-layout-5-col-2 ccm-layout-cell ccm-layout-col ccm-layout-col-2 last" style="width: 28.99%">…</div>
```

Styles   Computed   Event Listeners   »

```
element.style {
}
```

```
media="screen"                    www.granneman.com/
p, dl, li, th, td, blockquote    typography.css:114
{
  font: ▶16px/1.5em "Source Sans Pro", "Lucida
        Grande", Lucida, Verdana, sans-serif;
  color: ■#333;
}
```

```
media="screen"                    www.granneman.com/
p, blockquote, ul, ol, dl,       typography.css:109
fieldset, address, pre {
  margin-bottom: 15px;
  color: ■#333;
}
```

...   #not_footer   #page   #page_content   #body_copy   #ccm-layout-wrapper-14   #ccm-layout-body-5-2   div   div   div   div   **p**

Validation

CSS Validation Service

Check Cascading Style Sheets (CSS) and (X)HTML documents with style sheets

By URI   **By file upload**   **By direct input**

Validate by file upload

Choose the document you would like validated:

Local CSS file: [Choose File] no file selected

▸ **More Options**

jigsaw.w3.org/css-validator/

Try now the W3C Validator Suite™ premium service that checks your entire website and evaluates its conformance with W3C open standards to quickly identify those portions of your website that need your attention.

The W3C validators are developed with assistance from the Mozilla Foundation, and supported by community donations.
Donate and help us build better tools for a better web.

4627

Flattr

Note: If you want to validate your CSS style sheet embedded in an (X)HTML document, you should first check that the (X)HTML you use is valid.

Color Pickers

Sip

Color picker

theolabrothers.com

$0 (with $9.99 in-app purchase)

sip

Color History
Sea and Sky
Colors for Sip website
Kim's Project
Magic Hour
Another Palette

Start adding colors...

UIColor RGB

Magic Hour

Blue Bell
RGBA (0.65, 0.62, 0.83, 1.00)

Trendy Pink
RGBA (0.49, 0.38, 0.52, 1.00)

Cream Can
RGBA (0.92, 0.75, 0.35, 1.00)

Purple Heart
RGBA (0.44, 0.17, 0.69, 1.00)

Palettes

Sea and Sky

Colors for Sip website

Kim's Project

Magic Hour

Another Palette

You Can start adding colors...

The best way to collect, organize & share your colors.

Download **Sip for Mac**

Download **Sip for iPhone**

Sip **Pro**

49 Color Formats

and it keeps growing. From: CSS, NSColor, Android, Java, UIColor and more!

Keyboard Shortcuts

Sip makes your work process faster with tasty shortcuts. Here's the list.

Multi Touch Gestures

Using Multi touch gestures works just as you expect it would. See it in action

Share Color by Email

Sip makes it really easy to share any color with your friends. Here's how.

Drag Color

Drag and drop a color to and from any native application you are using. Here's how.

Illustrator Extension

Want to quickly add any color on your screen to the illustrator Tool panel? Done!

Photoshop Extension

Add colors to your Foreground and background "color well". See how.

Palettes

PRO

A new way to create, organize, edit and share colors with everyone.

Multiple Colors

PRO

Easy as it should be. Just press shift and click in the sequence of colors you need.

Edit Color

PRO

A quick and simple way to Edit Colors without the need of any additional software.

Custom Formats

PRO

Customize the color formats already provided, and also create and share new ones.
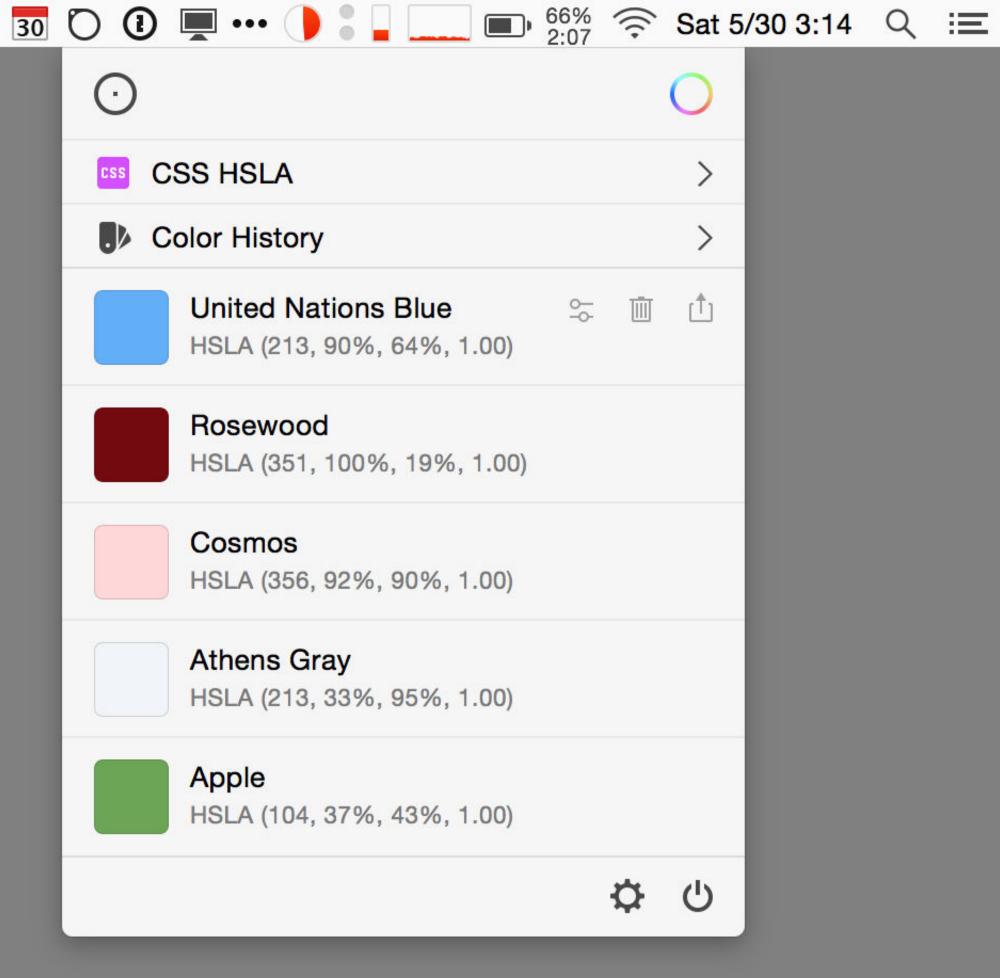
Snapshots

PRO

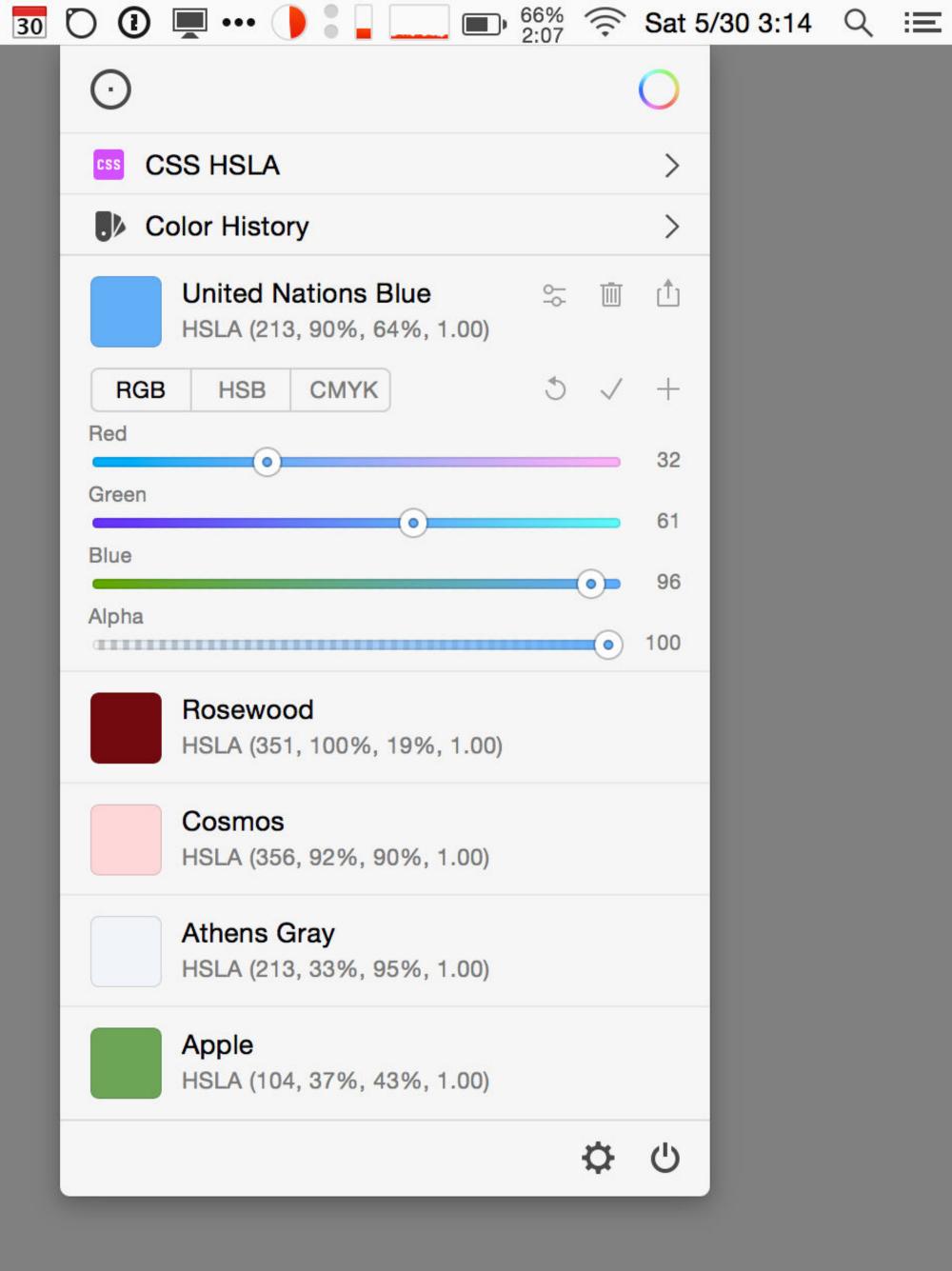Sip Pro saves you daily Snapshots. You can also save one in any specific moment.

Sip Everywhere

PRO

Besides all the new features, you now have all content synced in your devices.

CSS HSLA

Color History

United Nations Blue
HSLA (213, 90%, 64%, 1.00)

Rosewood
HSLA (351, 100%, 19%, 1.00)

Cosmos
HSLA (356, 92%, 90%, 1.00)

Athens Gray
HSLA (213, 33%, 95%, 1.00)

Apple
HSLA (104, 37%, 43%, 1.00)

Sat 5/30 3:14

CSS HSLA   >

Color History   >

United Nations Blue
HSLA (213, 90%, 64%, 1.00)

RGB   HSB   CMYK

Red
                                                           32
Green
                                                           61
Blue
                                                           96
Alpha
                                                          100

Rosewood
HSLA (351, 100%, 19%, 1.00)

Cosmos
HSLA (356, 92%, 90%, 1.00)

Athens Gray
HSLA (213, 33%, 95%, 1.00)

Apple
HSLA (104, 37%, 43%, 1.00)

Android ARGB
Android RGB
Android XML

CGColor CMYK
CGColor RGB

CSS Hex

CSS HSL
✓ CSS HSLA
CSS RGB
CSS RGBA

Generic CMYK 8-Bit
Generic CMYK Decimal
Generic CMYK Hex
Generic RGB 8-Bit
Generic RGB Decimal
Generic RGB Hex

Java RGB
Java RGBA

.Net RGB
.Net ARGB

NSColor Calibrated HSB
NSColor Calibrated RGB
NSColor Device CMYK
NSColor Device HSB
NSColor Device RGB
NSColor HSB
NSColor RGB

CSS HSLA   >

Color History   >

United Nations Blue
HSLA (213, 90%, 64%, 1.00)

Rosewood
HSLA (351, 100%, 19%, 1.00)

Cosmos
HSLA (356, 92%, 90%, 1.00)

Athens Gray
HSLA (213, 33%, 95%, 1.00)

Apple
HSLA (104, 37%, 43%, 1.00)

Just Color Picker

Small & with the necessary features

$0

annystudio.com/software/colorpicker/

Just Color Picker

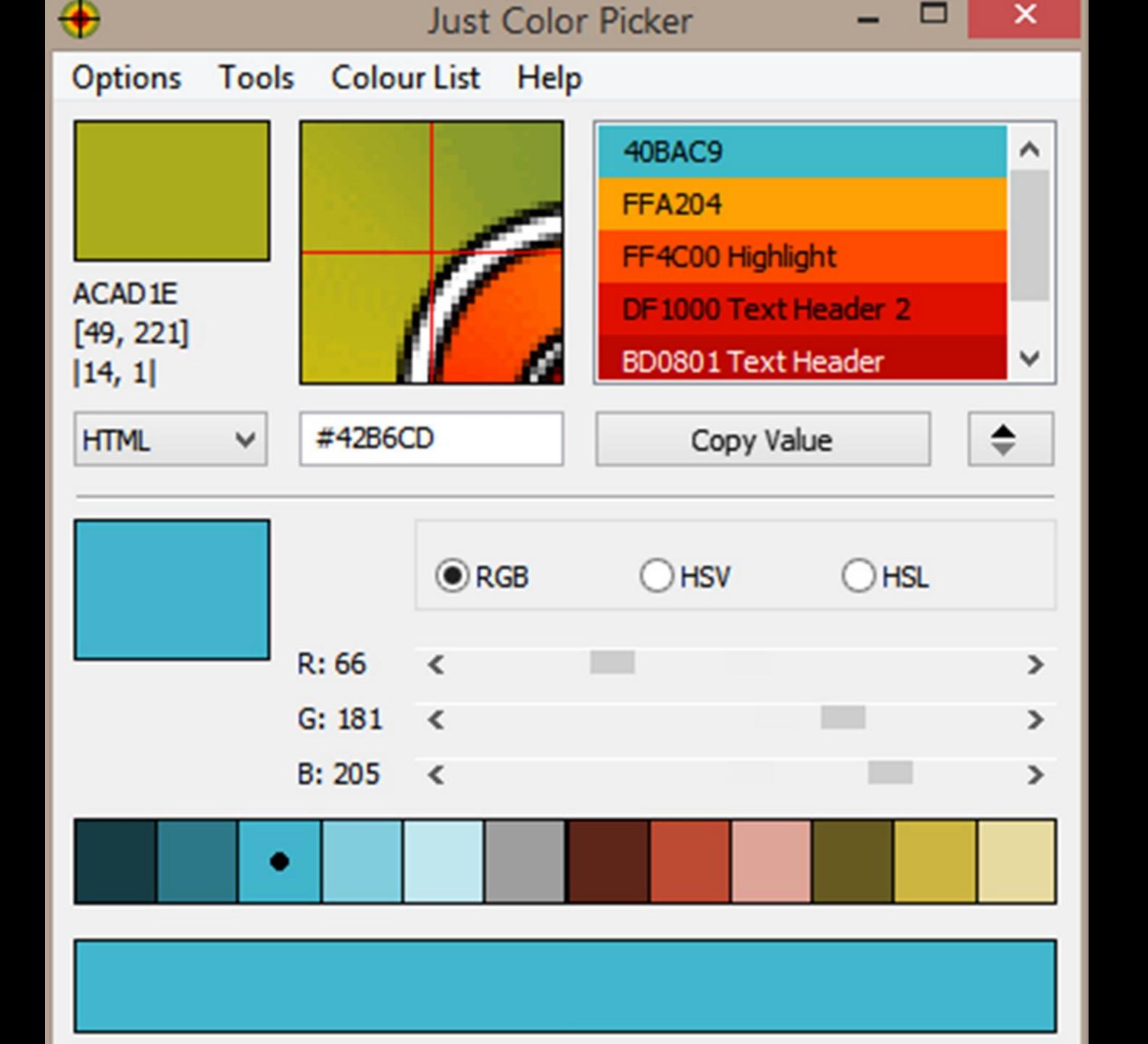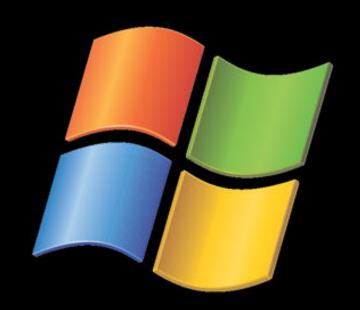## Free portable colour picker and colour editor for web designers and digital artists

Quick links: *Download, Online Manual, Q&A, Version History*

While taking part in any computer routine, graphic designers, digital artists and web-designers may see colours and colour combinations they would like to use in their projects. With Just Color Picker, identifying the colours, saving, editing and combining them into beautiful colour combinations is a quick and simple job.

One of the main advantages of Just Color Picker is that it was created by a person who uses it on an everyday basis. It is a software for designers and digital artists created by a designer and digital artist.

Just Color Picker

Options   Tools   Colour List   Help

40BAC9
FFA204
FF4C00 Highlight
DF1000 Text Header 2
BD0801 Text Header

ACAD1E
[49, 221]
|14, 1|

HTML

#42B6CD

Copy Value

○ RGB   ○ HSV   ○ HSL

R: 66

G: 181

B: 205

ColorPro

Professional color picker

www.iconico.com/colorpro/

$30

ICONICO

ColorPro

The Ultimate HTML Color Picker for Professionals

ColorPro Home

Download — Try before you buy

Buy $29.50

Tips on Getting Started

Palette Export Formats

Help

Movies

ColorPro Forum

ColorPro 2.6 Released

Stop Minimize to System Tray

Saving Various Settings

Selecting Chips, Rearrange @

http://www.iconico.com/color is missing the images

Unable to close application: "Access is denied" error

Edit a color

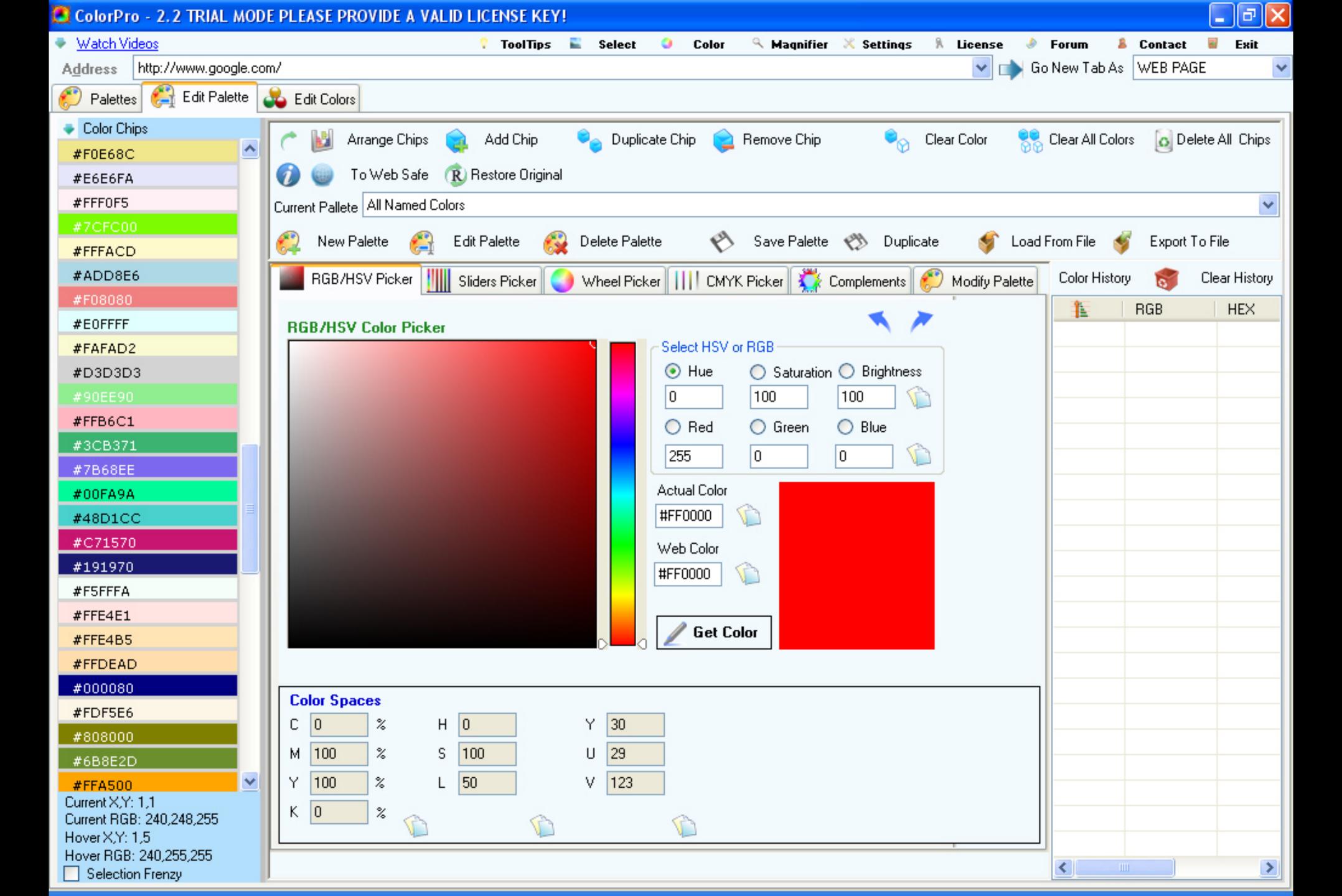ColorPro 2.4 Released

ColorPro

At Iconico we've offered our free ColorPic for years. It's so popular that it is the number one downloaded colorpicker on the web, however we thought it was time we went 'Pro'.

ColorPro has the same great color manipulation abilities, but with so much more. We've added webpage and image color extraction and manipulation, and advanced palette management, import and export. Once you go Pro you won't go back.

A Spectrum of Possibilities For Web Designers

One of the best parts about designing a web page is selecting the color schemes that will reach out and really grab the attention of your audience. In the past, a web designer would have to produce several different sets of the same graphic, all in different colors, just to test out different palette variations or to meet the branding needs of several clients. This would involve many hours of HTML coding, and researching and recording the various hex codes for all of the different colors that you needed to use.

Watch Videos

ToolTips | Select | Color | Magnifier | Settings | License | Forum | Contact | Exit

Address http://www.google.com/

Go New Tab As | WEB PAGE

Palettes | Edit Palette | Edit Colors

Color Chips

| #F0E68C |
| #E6E6FA |
| #FFF0F5 |
| #7CFC00 |
| #FFFACD |
| #ADD8E6 |
| #F08080 |
| #E0FFFF |
| #FAFAD2 |
| #D3D3D3 |
| #90EE90 |
| #FFB6C1 |
| #3CB371 |
| #7B68EE |
| #00FA9A |
| #48D1CC |
| #C71570 |
| #191970 |
| #F5FFFA |
| #FFE4E1 |
| #FFE4B5 |
| #FFDEAD |
| #000080 |
| #FDF5E6 |
| #808000 |
| #6B8E2D |
| #FFA500 |

Current X,Y: 1,1
Current RGB: 240,248,255
Hover X,Y: 1,5
Hover RGB: 240,255,255
☐ Selection Frenzy

Arrange Chips | Add Chip | Duplicate Chip | Remove Chip | Clear Color | Clear All Colors | Delete All Chips

To Web Safe | Restore Original

Current Pallete | All Named Colors

New Palette | Edit Palette | Delete Palette | Save Palette | Duplicate | Load From File | Export To File

RGB/HSV Picker | Sliders Picker | Wheel Picker | CMYK Picker | Complements | Modify Palette

Color History | Clear History

RGB/HSV Color Picker

Select HSV or RGB

◉ Hue   ◯ Saturation   ◯ Brightness
0   100   100

◯ Red   ◯ Green   ◯ Blue
255   0   0

Actual Color
#FF0000

Web Color
#FF0000

Get Color

	RGB	HEX

Color Spaces

C	0	%	H	0	Y	30
M	100	%	S	100	U	29
Y	100	%	L	50	V	123
K	0	%				

Thank you!

scott@granneman.com
www.granneman.com
ChainsawOnATireSwing.com
@scottgranneman

jans@websanity.com
websanity.com

CSS Overview

Selectors, Integration, Inheritance, Cascading

R. Scott Granneman & Jans Carton

2.4

Changelog

2020-07-31 2.4: Created diagram for the *Browser Processing Pipeline*; made *Descendant Combinator* examples clearer by adding arrows

Changelog

2020-07-21 2.3: Added list of Web APIs after JavaScript; added slides about design patterns in *Formatting*; added example of looking up property's initial value at MDN; updated screenshot for `initial` value & added explanation to it; added additional, easier examples for descendant & child combinators; changed *CSS Resets* to *CSS Reset Keywords*; improved wording explaining `inherit` value; moved *Key Selectors* at the beginning of *Complex Selectors*

Changelog

2020-07-15 2.2: Minor fixes; re-did *Specificity* completely; removed the *Miller's Crossing* example; made clearer in *Order* what is in HTML & what is in CSS

2020-07-10 2.1: Added note about Safari bug with `all: unset`

Changelog

2020-07-09 2.0: (con't. from ↓) added better descriptions of `<span>` & `<div>`; changed "Default inherited properties" to "Partial list of inherited properties" & added `text-edge`; added detail to MDN's CSS Reference in *Tools*; updated screenshots of CSS-Tricks in *Tools*; added CSS Resets section for `initial`, `inherit`, `unset`, `revert`, & `all`; added explanation of CSS keywords; so many changes I bumped version number up to 2!

Changelog

2020-07-09 2.0: Added more details to Specificity; minor edits; added definition of simple selector; moved *Key Selectors* under *Complex Selectors*; for user styles, gave more detail re: Firefox's `userContent.css`, Chrome, & extensions, & updated screenshot of Safari's Advanced Preferences; better screenshot for embedded styles; updated screenshot for `.class` selector; updated table showing difference between HTML & CSS terms; added order Jans places things in his stylesheets; (con't. ↑)

Changelog

2018-12-06 1.20: Added screenshots showing embedding with `<style>` & how to link to `main.css`; added logos for MDN supporters; `*:first-child` is the same as `:first-child`; added Side Note re: using 2 `class` attributes; in Formatting, always put `;` at the end of each declaration; added CSS Tricks to Tools; in Class, told viewer to see Compound to learn how to match an element with 2 classes; screenshots for `@import` & reorganized those slides; minor wording changes; better example for Compound selectors; replaced ID screenshots

Changelog

2018-11-21 1.19: Called out items on browser processing pipeline illustration; fixed image for multiple classes (`<figurecaption>`?!); updated screenshots for child combinator; for selector lists, removed silly example & added headings slide; added Side Note on `<div>` that it should semantically be `<ol>`; fixed Cascade example so proper method is shown; added slides showing difference between descendant & child combinators; added table on Time & User Perception to `@import`; replaced `inherit` screenshot

Changelog

2018-10-01 1.18: Added how DOM changes attribute-value pairs; replaced DOM section with The Browser Processing Pipeline; fixed Default inherited properties slide; updated theme to Granneman 1.5; added Walt Whitman on contradicting yourself

2017-11-06 1.17: Improved key selectors; added reasons for user CSS

Changelog

2017-10-30 1.16: Added Just Color Picker; added screenshot of ColorPro website; changed color of some arrows & shapes to Tulip Tree (`#E8A433`); better solutions to specificity problem with `.callOut`; fixed wording to be correct &/or more specific; added emoji for class names; added Opera user-agent styles; better examples for Key Selectors; applied Granneman 1.4 theme; fixed formatting issues; added default order I use in stylesheets

Changelog

2017-10-25 1.15: Added better examples for DOM vs source code; organized Basic Selectors much better; moved Key Selectors, IDs as page fragment identifiers & JavaScript hooks, & Compound Selector example from CSS Selectors to here; gave full list of selectors & grayed out ones we're not covering here; fixed wording introducing the Cascade

Changelog

2017-10-18 1.14: Took out details about how to turn on Firefox 3D View, since it's not longer supported; made Cascading chapter slide italicized; corrected & added info on anonymous boxes; in History, hid modules & added Can I Use, minor fixes & corrections

2016-09-23 1.13: Moved slide comparing class & ID; under Importance, rearranged IE & Safari & added Firefox for Windows, Chrome, & Edge; re-ordered examples of Descendant Combinator; fixed formatting errors

Changelog

2016-09-16 1.12: Updated theme to Granneman 1.2; small changes in wording to make things clearer; cleaned up formatting in a few places; added slide re: using classes instead of IDs; fixed slides in Selector Grouping; changed Important example from WordPress to Bootstrap; fixed wrong information re: `class` & `id` values & clarified; added example for Descendant Combinator

Changelog

2016-01-20 1.11: Added slide re: CSS3 Taxonomy & Status; better explanation why we need `<div>`; added note re: specificity

2016-01-11 1.10: Minor improvements taken from CSS - Selectors; added Child Combinator to Selectors; added another example of Child Combinator; added screenshots of browser CSS; explained author vs author in Importance; explained how my Safari CSS works; added a long example showing how the Cascade works in practice

Changelog

2015-12-13 1.9: Clarified source of DOM quote; changed numbers of selectors; got rid of `E` & `F` in selectors & made them clearer; changed `.bigRed` to `.big-red`; clarified source of class & ID names; add tweet re: CSS to beginning

2015-05-10 1.8: Added info about CSS 4; clarified that `<span>` & `<div>` draw boxes; added additional names of directories that are always created; changed "What Google prefers" to "… uses"; removed Hues & added Sip to Color Pickers; fixed URL & screenshots for CSS Vocabulary; moved Viewport Resizer to Bootstrap; for Separation of Concerns, added "& Meaning" to HTML

Changelog

2015-03-06 1.7: Added another example of selector grouping; added details about *resource inlining*

2015-01-12 1.6: Added my `safari.css` file; clarified & added info on specificity

2015-01-11 1.5: Clarified Inheritance

2014-09-27 1.4: Changed "browser" to "rendering engine" in a few places where it made sense

Changelog

2014-08-12 1.3: Improved Descendant Selector examples; improved wording & added slides in DOM section; improved Cascade diagram; fixed `<div>` screenshot; added URLs for `<div>` & `<span>` screenshots; fixed Viewport Resizer screenshots

2014-08-10 1.2: Added DOM spec info & screenshots of DOM & Source Code; added details about Firefox Web Dev Tools

2014-08-04 1.1.1: Added definition of *anonymous object*

Licensing of this work

This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License.

To view a copy of this license, visit http://creativecommons.org/licenses/by-sa/4.0/.

You are free to:

» *Share* — copy and redistribute the material in any medium or format
» *Adapt* — remix, transform, and build upon the material for any purpose, even commercially

Under the following terms:

Attribution. You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use. Give credit to:

Scott Granneman • www.granneman.com • scott@granneman.com

Share Alike. If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.

No additional restrictions. You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

Questions? Email scott@granneman.com