

Cascading Style Sheets

Formatting Using Standards

Washington University in St. Louis

R. Scott Granneman

scott@granneman.com

www.granneman.com

© 2002-2009 R. Scott Granneman

Last updated 20091114

You are free to use this work, with certain restrictions.

For full licensing information, please see the last slide/page.

History

CSS 1
December 1996

CSS 2
May 1998

CSS 2.1
July 2007

CSS 3
Under development

				
	Trident	Gecko	WebKit	Presto
CSS 1	7.0+	1.0+	85+	7.0+
CSS 2.1	Mostly	Mostly	Mostly	Mostly
CSS 3	Slight	Partial	Partial	Partial

Source: [http://en.wikipedia.org/wiki/Comparison_of_layout_engines_\(Cascading_Style_Sheets\)](http://en.wikipedia.org/wiki/Comparison_of_layout_engines_(Cascading_Style_Sheets))

Selector	IE 5.5	IE 6	IE 7	IE8 as IE7	IE8 as IE8	FF 2	FF 3.0	FF 3.5	Saf 3.0 Win	Saf 3.1 Win	Saf 4.0 Win	Chrome 1	Chrome 2	Opera 9.62	Opera 10b	Konqueror 3.5.7
CSS 2	incorrect	incomplete	yes	yes	to be tested	yes	yes	to be tested	yes	to be tested	yes	yes	yes	to be tested	to be tested	yes
<p>CSS 2 has become the baseline of CSS support; without it a browser is decidedly backward.</p>																
CSS 3	minimal		incomplete	almost	to be tested	incomplete	almost	to be tested	almost	almost	almost	almost	almost	to be tested	to be tested	almost
<p>The compatibility information here is about the CSS3 modules I test. It is not necessarily valid for the browsers' entire CSS3 support.</p>																

Source: <http://www.quirksmode.org/compatibility.html>

Why CSS?

Separate content from presentation

Site-wide consistency

Apply different styles
to same content
in different media:
Web browser, cell phone,
auditory, print, projector ...

Adherence to standards

It's fun!

Basic Selectors

HTML: <element attribute="value">

CSS: selector {property:value;}

CSS examples:

```
p {font-size: 12px;}
```

```
.attention {color: red;}
```

```
#footer {font-family: Verdana, Arial,  
          sans-serif;}
```

Don't do this:

```
p {color: red;}
```

```
p {font-size: 12px;}
```

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

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

Instead ...

Combine related property/value pairs:

```
p {  
    font-family: Verdana, Arial, sans-serif;  
    font-weight: bold;  
    font-size: 12px;  
    color: red;  
}
```

14 different selectors

Basic selectors:

- ✓ Type
- ✓ Class
- ✓ ID
- ✓ Selector grouping
- ✓ Universal selectors
- ✓ Descendant selectors

Type Selectors

Type selectors,
AKA HTML Element Selectors

Used when you want to affect every instance of an element

Example:

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

Result: text in every `<p>...</p>`
is red & bold

Advantages:
changes every instance
of the selected HTML element

Disadvantages:
changes every instance
of the selected HTML element

You're going to use
HTML element selectors a lot,
but use with caution!

Class Selectors

Used when you want to apply a class to any appropriate element, as many times per page as you desire

CSS (note the dot!):

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

HTML results

(note the lack of a dot!):

```
<p class="attention">...</p>
```

is red & bold

but `<p>...</p>` is not

CSS (note the dot!):

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

```
<p class="attention">...</p>  
is red & bold
```

```
<hr class="attention">  
does nothing
```


Element Class Selectors

What if you want a class
to only work
inside a particular HTML element?



In CSS:

```
p.title {  
  font-weight: bold;  
  text-align: center;  
  font-size: 16px;  
  font-family: "Times New Roman", serif;  
}
```

In HTML:

```
<p class="title">Linux Phrasebook</p>
```

Rules for CSS class names

	
a-z	Space
A-Z	Starting with #
0-9	Starting with -#
-	
_	

Describe function, not appearance!

Bad class names:

.author name

.#1_title

.bigRed

Good class names:

.authorName

.title

.emphasis

.footnote

.center

Multiple words in a class name?

These are all OK

.maincontentnav

.mainContentNav

.MainContentNav

Just be consistent!

ID Selectors

Used when you want to apply a ID
to any appropriate element,
only once per page

CSS (note the pound sign!):

```
#navigation {color: #0F0; font-size: 14px;}
```

HTML (note the lack of a pound sign):

```
<p id="navigation">...</p>
```



is green & 14px

```
<p>...</p>
```

 is not

Remember: only once per page!

Rules for CSS ID names

	
a-z	Space
A-Z	Starting with #
0-9	Starting with -#
-	
_	

Describe function, not appearance!

Bad ID names:

#top nav

#tinyLittleFonts

Good ID names:

#sidebar

#mainNav

#footer

#logo

Multiple words in a ID name?

These are all OK

#maincontentnav

#mainContentNav

#MainContentNav

Just be consistent!

Selector Grouping

In CSS:

```
p, blockquote, td, option {  
  font-family: Verdana, Arial, sans-serif;  
  font-size: 12px;  
}
```

In HTML:

```
<p>Place text here.</p>
```

```
<blockquote>Place text here.</blockquote>
```

Results

- ✓ Specific, defined fonts
- ✓ Unified look & design

Universal Selectors

* {font-family: Verdana, sans-serif;}

* matches

every (appropriate) HTML element

Instead of this

```
p, blockquote, td, option {  
  font-family: Verdana, Arial, sans-serif;  
  font-size: 12px;  
}
```

Use this

```
* {  
  font-family: Verdana, Arial, sans-serif;  
  font-size: 12px;  
}
```

The Global White Space Reset

```
* {  
  margin: 0;  
  padding: 0;  
}
```


Descendant Selectors

```
td p {font-size: 10px;}
```

Any `<p>` inside a `<td>` will be 10px

`<p>` can be a child, grandchild,
or later descendant

Any other `<p>` is unaffected

Used to be called
Contextual Selectors
before renaming to
Descendant Selectors

&

<div>

HTML elements
“work”
without attributes & values

(except `img` & `a`)

`` & `<div>`
are HTML elements
designed solely as selectors for CSS

`` & `<div>` by themselves
do nothing on a Web page

They *must* have
a CSS-based attribute/value pair
(`class="x"` or `id="x"`)
to do anything productive

`` is an inline element
that does nothing without CSS

Use ``
to hold attribute/value pairs
relevant to CSS

```
<p>  
  My name is <span  
class="emphasis">Borat</span>!  
</p>
```


`` is tedious

You can instead use

`...` or `...`

Virtually all browsers
make `` bold & `` italic

You can't assume that, however,
so you need to do this in your CSS

```
em {font-style: italic;}  
strong {font-weight: bold;}
```

<div>

`<div>` is a block-level element
that does nothing without CSS

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

```
<div class="navigation">  
  <p class="off">...</p>  
  <p class="on">...</p>  
  <p class="off">...</p>  
</div>
```

Another <div> example

```
<div class="sidebar">  
  <p class="sidebarTitle">Sidebar Title</p>  
  <p class="sidebarText">  
    This is some text inside a sidebar. It  
    is <span class="emphasis">important  
    </span> that you understand span &  
    div, or you are <span class="emphasis">  
    doomed</span>!  
  </p>  
</div>
```

Integrating CSS

1. Inline styles
2. Embedded styles
3. Linking to external styles
4. @import

Inline

Before:

```
<p><font face="Times New Roman, Times, serif" size="12px" color="red">My name is Borat!</font></p>
```

After, using the style attribute:

```
<p style="font-family: 'Times New Roman', Times, serif; font-size: 12px; color: red;">My name is Borat!</p>
```


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

Use inline styles
for unique instances

Embedded

Place `<style>...</style>` in `<head>`

```
<head>
  <style type="text/css">
    p {
      font-family: Verdana, sans-serif;
      font-size: 12px;
    }
    .emphasis {font-weight: bold;}
  </style>
</head>
```

Embedded styles
are great for one page ...

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

Linking

Create an external style sheet:
a text file ending in “.css”

Link to external style sheet
on all pages in site

```
<head>  
  <link rel="stylesheet" type="text/css"  
href="path/to/stylesheet.css">  
</head>
```

What should you name
your CSS file?

main.css

client.css

search.css

navigation.css

Where should you place
your CSS file?

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

images

scripts

styles

Sample style sheet, “colostomo.css”:

```
/* CSS for Colostomo Inc.  
Created 27 Sept. 2002  
Last modified 4 Nov. 2006 by RSG */  
  
/* Common */  
p, blockquote, td {font-family: Verdana,  
Arial, sans-serif; font-size: 12px;}  
#footer {font-size: 10px;}  
.emphasis {font-weight: bold;}  
  
/* Navigation */  
.on {font-size: 14px; font-weight: bold;}  
.off {font-size: 14px;}
```

It's just a text file!

Use comments in CSS
for the same reason as in HTML

- ✓ Notes to yourself & others
 - ✓ Debug: comment out troublesome CSS for testing

HTML comments:

```
<!-- blah blah html html  
      blah blah html html -->
```

CSS comments:

```
/* blah blah css css blah blah css css */
```

If you have more than one page,
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

Break up your enormous CSS
into separate files
that are imported as needed

@import
can be used with
Embedded CSS or Linked CSS
@import must always come 1st

Embedded

```
<head>  
  <style type="text/css">  
    @import url("other.css");  
    p {font-family: Verdana, sans-serif;}  
  </style>  
</head>
```

Linked

```
/* colostomo.css */  
@import url("other.css");  
@import url("another.css");  
p {font-family: Verdana, sans-serif;}
```


Importance

CSS can come from 3 places:

The CSS author

The user

The browser

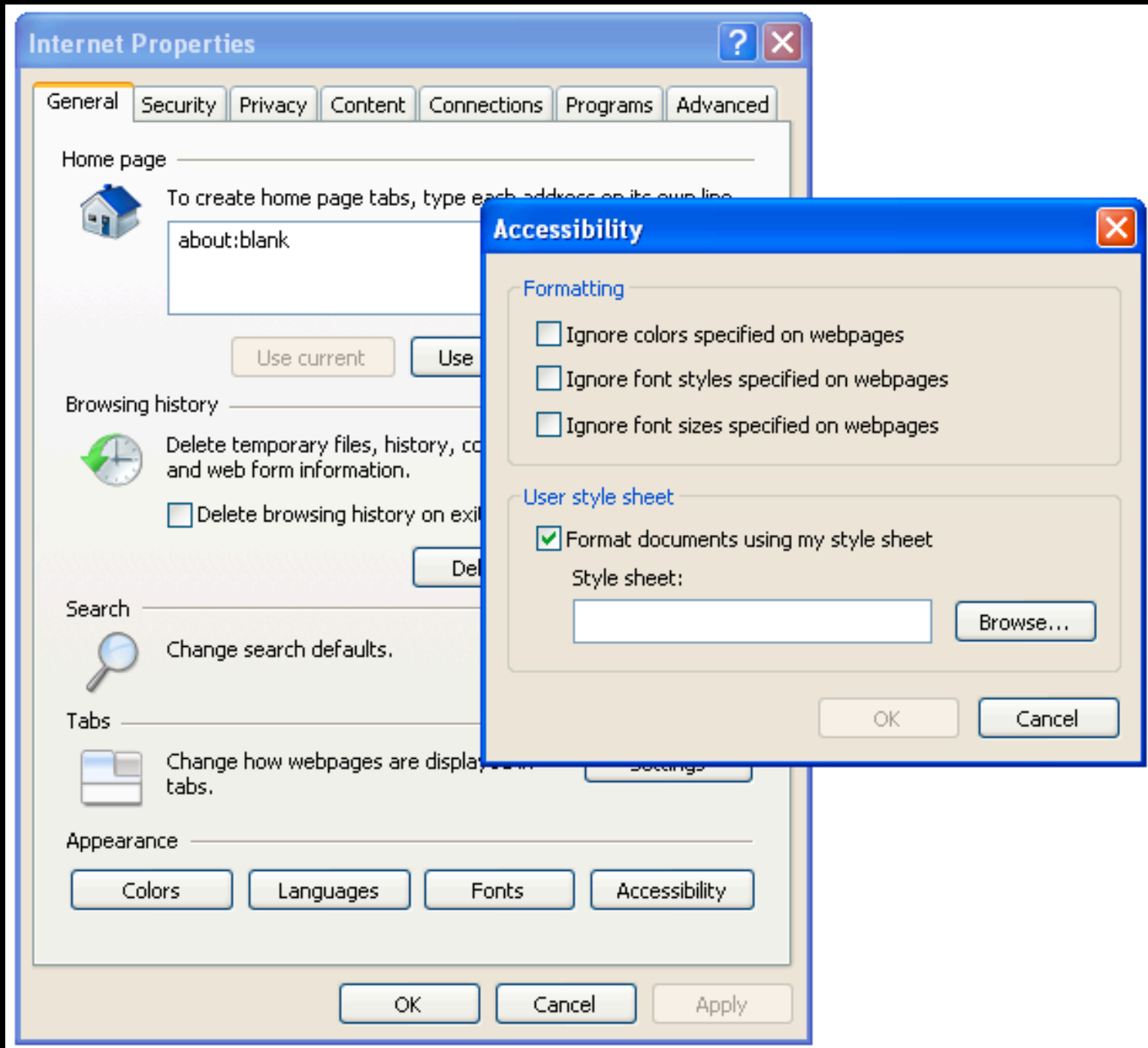
All Web browsers
have built-in CSS rules

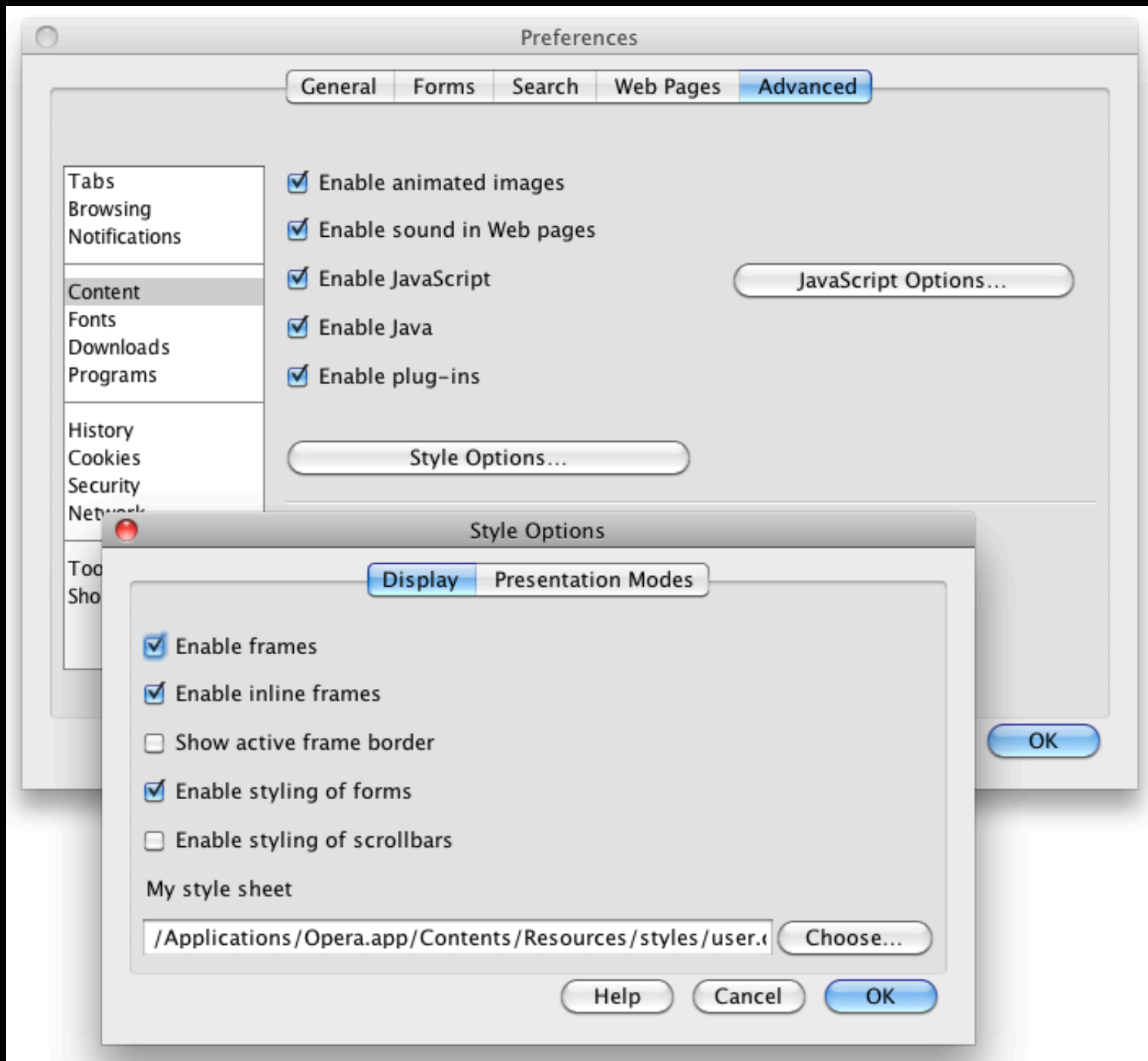
In Firefox,
why do `<p>`'s
have a certain amount of space
above & below them?

Firefox's built-in default CSS

Users can specify CSS rules too

If you have bad eyesight,
perhaps you want all fonts
to be a minimum size







Advanced



General



Appearance



Bookmarks



Tabs



RSS



AutoFill



Security



Advanced



Glims

Universal Access: Never use font sizes smaller than

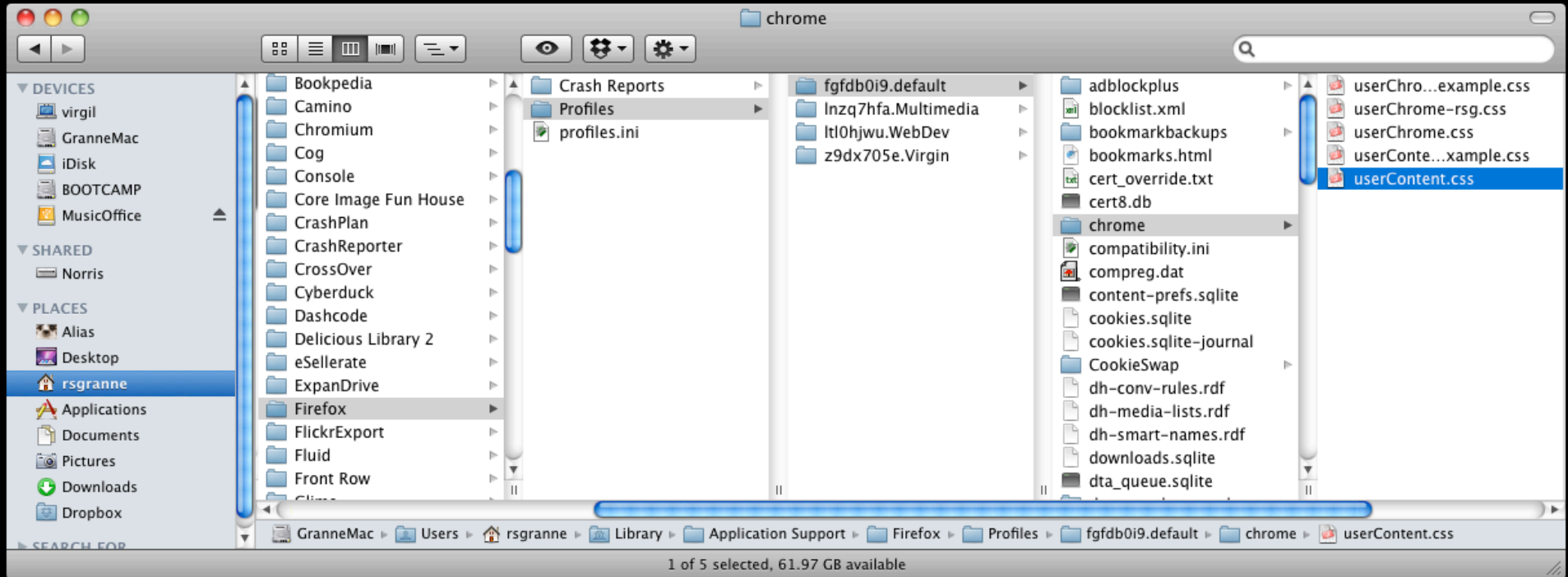
- Press Tab to highlight each item on a webpage
Option-Tab highlights all items except links.

Style sheet None Selected

Proxies Other...

- Show Develop menu in menu bar





If your user CSS contradicts
a site's CSS,
you can make sure yours "wins"

Use !important
after your property/value

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

CSS authors can do the same thing

The order of importance

AKA

The order in which stylesheets
are *weighted*

User !important



Author !important



Author



User



Browser

Inheritance

You have this CSS:

```
p {  
  font-family: "Times New Roman", serif;  
  font-size: 12px;  
}  
.emphasis {font-weight: bold;}
```

You have this HTML:

```
<p>
```

```
  And they called him <span  
  class="emphasis">Seamus</span>.
```

```
</p>
```

What does “Seamus” look like? Why?

A selector inside a selector inherits the outermost selector's property values, unless specifically overridden

C inside B inside A:

```
<a><b><c></c></b></a>
```

B inherits A

C inherits B & A

However ...

If A says font-size:12px
& C says font-size:10px,
then C will override A

The closest selector
being affected by the CSS wins

Cascading

How does the browser know
which CSS to apply to an element?

The rules follow a cascade

Read about it at
[http://www.w3.org/
TR/CSS2/cascade.html](http://www.w3.org/TR/CSS2/cascade.html)

Sample
Cascade
Rule:
The Last Wins

Later CSS wins
over earlier CSS

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

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

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

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

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

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

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

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

Cascade Rules

Specified CSS wins
over default browser CSS

Descendant selectors win
over HTML element selectors

Class selectors win
over descendant selectors

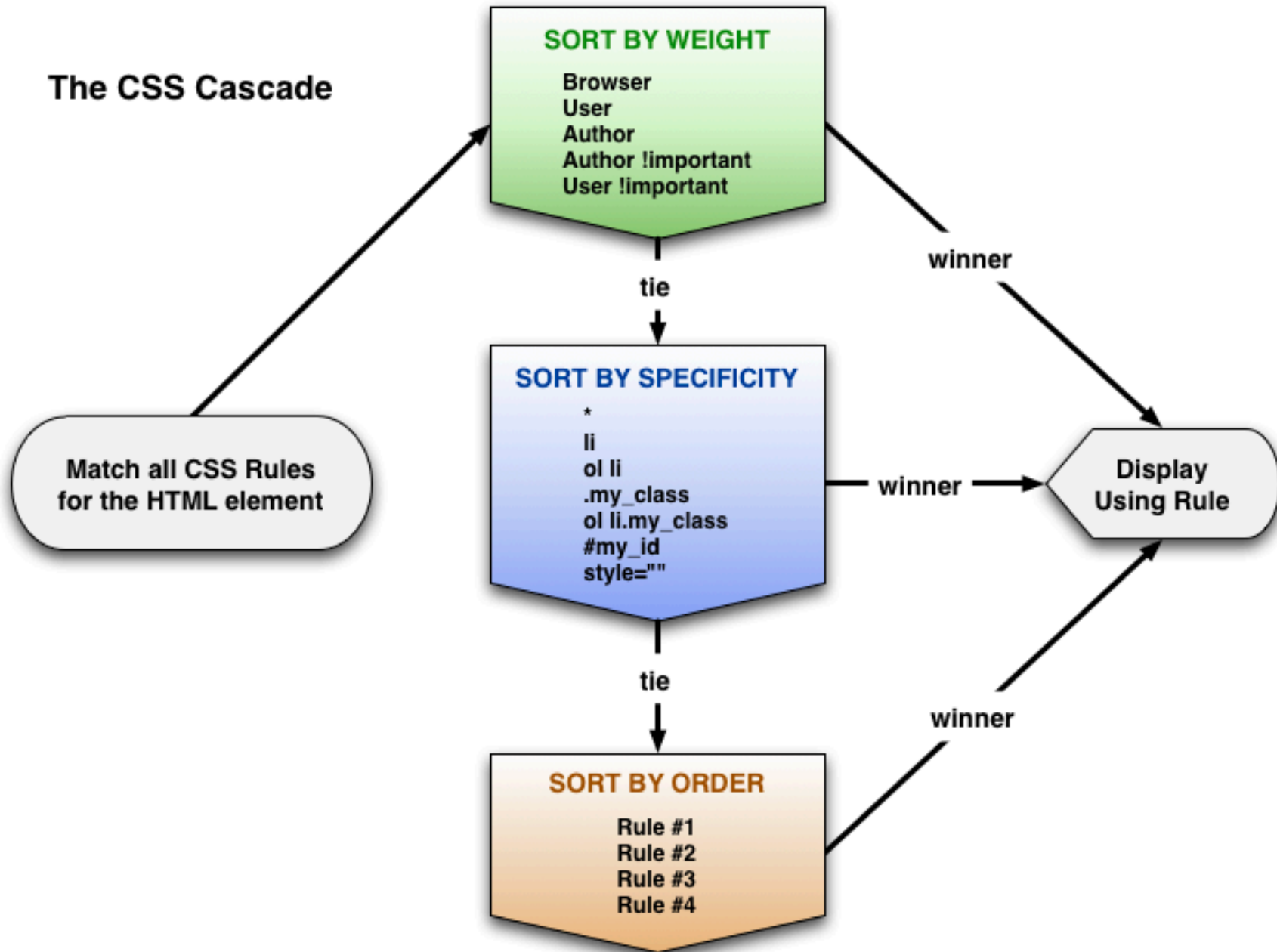
ID selectors win
over class selectors

Inline CSS wins over other CSS

Later CSS wins over earlier CSS

Cascade rules are applied in this order

The CSS Cascade



Tools

CSS Editors

TopStyle

- ✓ Windows only
- ✓ Lite (free) & Pro (\$80)
- ✓ Very good, but expensive

www.bradsoft.com

TopStyle Pro 3.0

File Edit Search Preview View HTML Tools Options Help

Class

index.asp index.asp style1.css

Tag Inspector

XHTML 1.0

Attributes

- class
- dir
- id
- lang
- style
- title

Deprecated Attributes

- alink
- background
- bgcolor
- link
- text
- vlink

Events

- onclick

Output

http://localhost/topstyle/index.asp

css/html/xhtmll editor for windows
Home Order Download Support Search About

topSTYLE

COMING IN JULY: TOPSTYLE PRO 3.0!

The web is evolving. It's time for a new approach to coding web pages.

Buy version 2.5 now and get 3.0 for free upon release!

Ln 21, Col 6

bradsoft.com

Style Master

- ✓ Windows & Mac: \$49
 - ✓ Some problems,
but great help & tutorials

www.westciv.com

screen.css [Macintosh HD:Users:johnallsopp:Sites:westciv 2:screen.css]

Statements Tools Preview Code Design Split Zoom Info Browsers Properties

Search only selectors

Statements

```

/* CSS reset thanks to eric meyer */
html, body, div, span, applet, object, ifram
:focus
body
ol, ul
table
caption, th, td
blockquote: before, blockquote: after, q: before
blockquote, q
body
#header
#header h1
#header h1 a: link, #header h1 a: visited
#header h1 a: hover
* html #header p
#metanav
#metanav ul li a: link, #metanav ul li a: visited
#metanav ul li a: hover
#header ul.sitesections
#header ul.sitesections li
#header ul.sitesections li a: feeds
#header ul.sitesections a
#header ul.sitesections a: hover
#header ul.sitesections a: hover, #header ul a:
.quicklinks
.quicklinks h1
.quicklinks ul
.quicklinks ul li
.quicklinks ul li a: link, .quicklinks ul li a: visit
.quicklinks ul li a: hover
.quicklinks p
#maintext
#maintext
#maintext a: link, .quicklinks p a: link
#maintext a: visited, .quicklinks p a: visited

```

16px, 115px

82px

tools & resources for web professionals

style master tools resources downloads

688px

About westciv

Read more about westciv.

Style Master

learn more about Style Master, our Mac and Windows CSS development tool.

Training

If you want to learn web development the right way, or improve your skills in HTML and CSS, then we've got the right training for you.

Tools

More about all k

Conferenc

Web Directio

web professiona

Resources

Keeping up with hard work. At w make your life e

© 1997- western civilisation Pty. Ltd. Contact Us Search Site

Property Editors

Advanced Style

text shadow

color # 000

v offset 1 px

h offset 1 px

blur 1 px

box shadow

color # bbb

v offset 1 px

h offset 1 px

blur 2 px

text stroke

color

width em

fill

opacity 0.9

Safari & Mozilla Compatibility

2 statements select this element

html > body > frontpage > div#header

index.html

Morphon CSS Editor

- ✓ Windows, Mac, & Linux: \$39

- ✓ Works well,
especially on a fast machine

- ✓ No longer updated,
but still usable

www.morphon.com

CSSEdit

- ✓ Mac only: \$45
- ✓ Very nice, useful software
macrabbbit.com/cssedit/

main.css

Style Group Comment New Selector Search

Export Preview Milestones Validate Editors

main.css

Styles

- Common
- a:link
- a:visited
- a:hover, a:active
- body, td, th, li, input, select, option
- body
- h1, h2, h3, h4, h5, h6
- h1
- h2
- h3
- h4
- h5
- h6
- h1 + p, h2 + p, h3 ... + p, h5 + p, h6 + p
- p
- textarea
- .achtung
- Page Header
- #siteName
- #siteName h1
- #siteName a
- #utilityNav
- #utilityNav a
- #utilityNav a:link, #utilityNav a:visited
- #utilityNav a:hover, #utilityNav a:active
- DHTML Menu
- #coreContentNav
- a.menuButton, ...uButton:visited
- div.dhtmlMenu
- div.dhtmlMenu a.dhtmlMenuItem
- div.dhtmlMenu a.dhtmlMenuItem:hover
- Path Navigation

```

1 /* Common */
2 a:link { color: #000099; }
3 a:visited { color: #660099; }
4 a:hover, a:active { color: #990000; }
5
6 body, td, th, li, input, select, option {
7     font-family: Candara, Verdana, Arial, Helvetica, sans-
8     serif;
9     font-size: 12px;
10 }
11 body {
12     margin: 0px;
13     padding: 0px;
14     background-color: #FFFFFF;
15 }
16
17 h1, h2, h3, h4, h5, h6 { font-family: Cambria, Georgia,
18 "Bitstream Vera Serif", "Times New Roman", Times, serif; }
19
20 h1 {
21     font-size: 26px;
22     font-weight: bold;
23     margin: 0px 0px 2px 0px;
24 }
25
26 h2 {
27     font-size: 22px;
28     font-weight: bold;
29     margin: 10px 0px 5px 0px;
30     clear:both;
31 }
32
33 h3 {
34     font-size: 18px;
35     font-weight: bold;
36     margin: 10px 0px 5px 0px;
37     border-bottom: 1px dotted grey;
38 }
39
40 h4 {
41     font-size: 16px;
42     font-weight: normal;
43     margin: 10px 0px 5px 0px;
44     font-style: italic;
45 }

```

Fonts & Color

Color:

Size: 12 pixels

Line Height:

Style: B B + i i None

Decoration: aa aa aa None

Variant: SMALL CAPS Normal

Font Family: Candara Verdana Arial

Text Background Dimensions Positioning & Page Flow Borders Bullets & Advanced

Browser Tools



Web Developer

<http://addons.mozilla.org/firefox/60/>

Firebug

[https://addons.mozilla.org/
firefox/1843/](https://addons.mozilla.org/firefox/1843/)

W3C CSS Validator

<http://jigsaw.w3.org/css-validator/>

Websites

Annotated collection of CSS links:

[http://www.granneman.com/
usefullinks/development/coding/css/](http://www.granneman.com/usefullinks/development/coding/css/)

My updated collection
of CSS bookmarks:

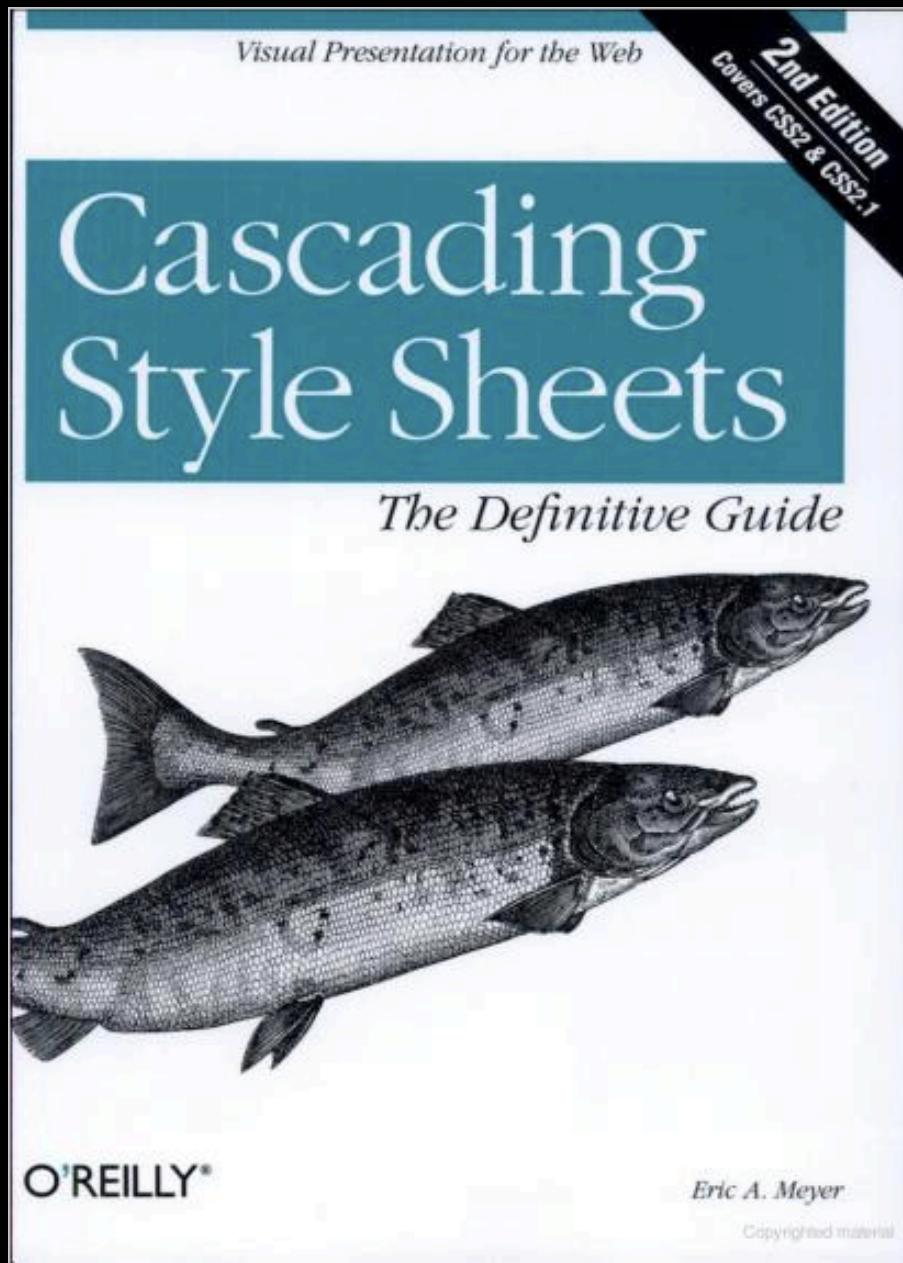
<http://del.icio.us/rsgranne/css>

QuirksMode
<http://www.quirksmode.org>

Books

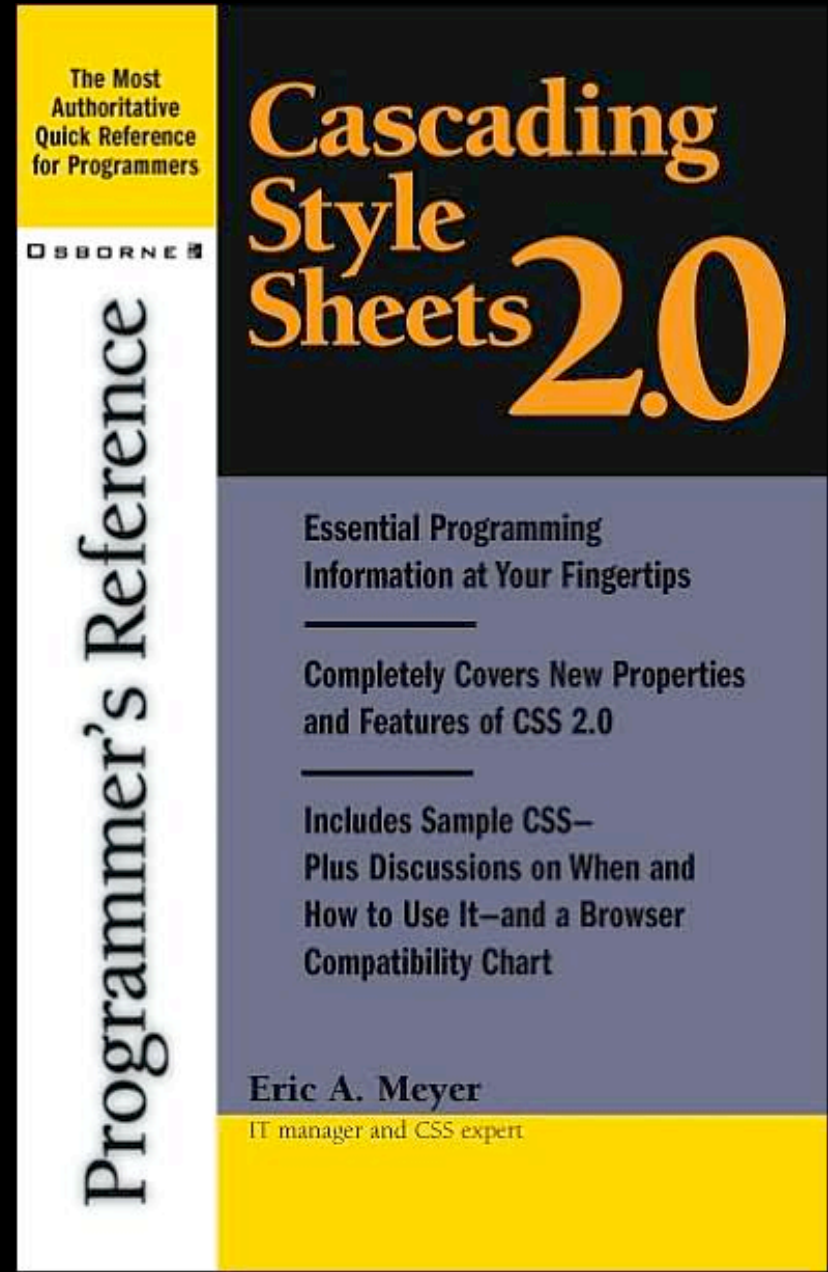
Eric Meyer, CSS god

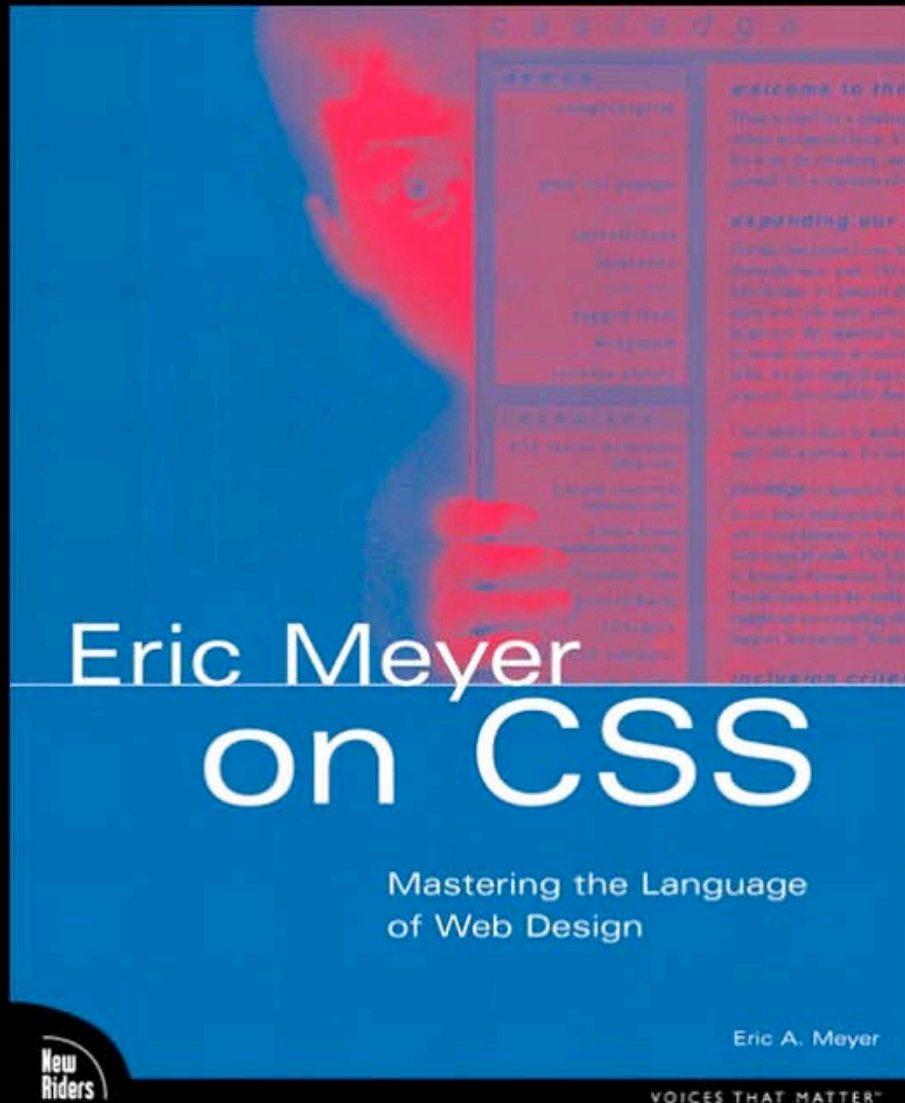




Detailed overview of CSS 1

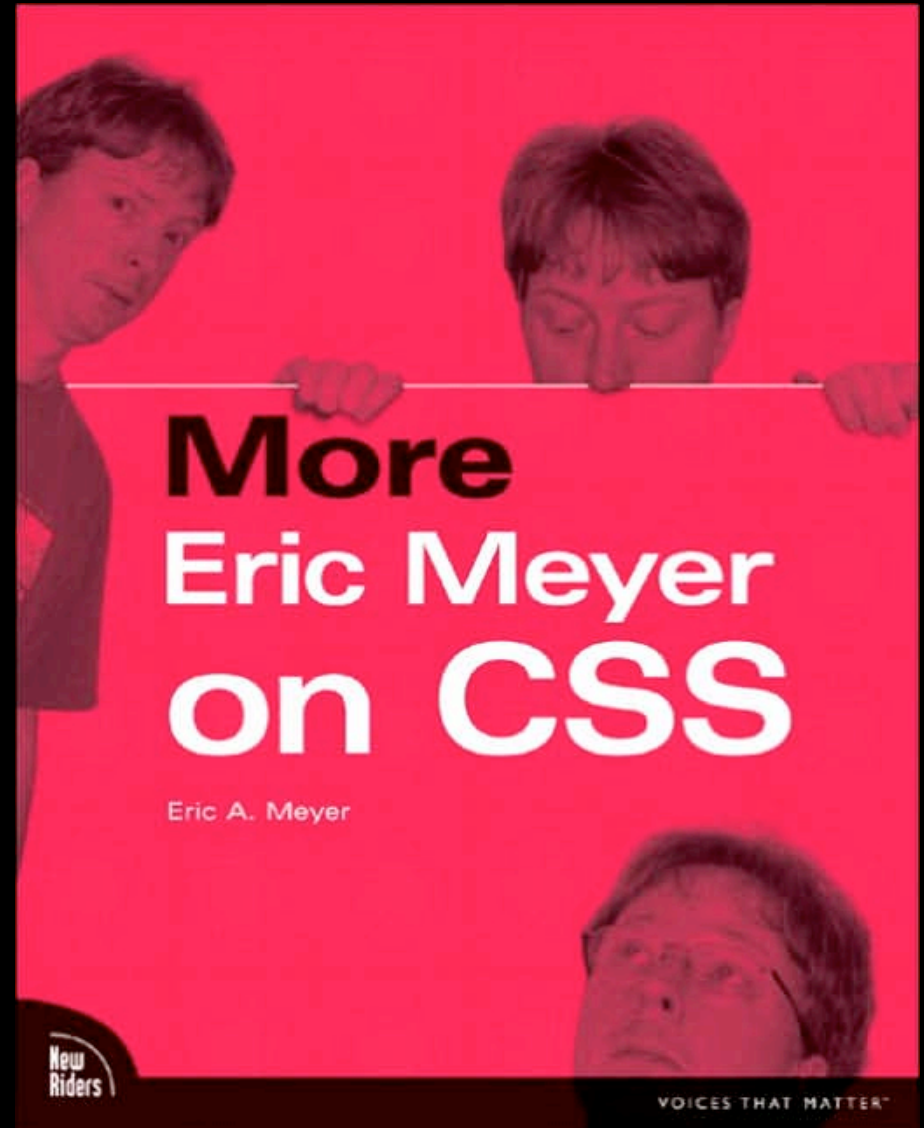
Quick ref guide,
with focus on CSS 2





Details 13 projects, showing how & why to use CSS

Details 10 projects,
showing how & why
to use CSS



Thank you!

Email: scott@granneman.com

Web: www.granneman.com

Publications: www.granneman.com/pubs

Blog: blog.granneman.com

Twitter: [scottgranneman](https://twitter.com/scottgranneman)

Cascading Style Sheets

Formatting Using Standards

Washington University in St. Louis

R. Scott Granneman

scott@granneman.com

www.granneman.com

© 2002-2009 R. Scott Granneman

Last updated 20091114

You are free to use this work, with certain restrictions.
For full licensing information, please see the last slide/page.

Licensing of this work

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

To view a copy of this license, visit

<http://creativecommons.org/licenses/by-sa/1.0>

or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

In addition to the rights and restrictions common to all Creative Commons licenses, the Attribution-ShareAlike License features the following key conditions:

Attribution. The licensor permits others to copy, distribute, display, and perform the work. In return, licensees must give the original author credit.

Share Alike. The licensor permits others to distribute derivative works under a license identical to the one that governs the licensor's work.

Questions? Email scott@granneman.com