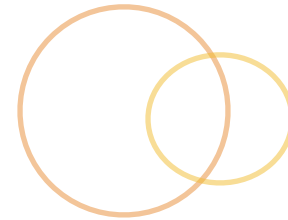


CSS: Introduction



Web Development



Copyright 2003 - 2008 DevelopIntelligence LLC
<http://www.DevelopIntelligence.com>

CSS: Beauty & the Beast



⦿ The Beast

- ⦿ (X)HTML - U.G.L.Y. you ain't got no alibi
- ⦿ It is the structure to our web document
- ⦿ it is the bones, muscles, fat, chest hair
- ⦿ not much to modify appearance wise (lists, paragraphs, headings)
- ⦿ organize data well - make dress up time easier

⦿ The Beauty

- ⦿ CSS - Cascading Style Sheets
- ⦿ Makes things look good
- ⦿ The clothes, the jewelry, the bling
- ⦿ centralizes control of appearance over whole website

CSS: Beauty & the Beast



◎ The Beauty

- ◎ CSS - Cascading Style Sheets
- ◎ layout columns, overlay text on images
- ◎ line-spacing, text-indenting, font properties, margins, borders, list styles, background properties ...
- ◎ used to layout page instead of Legacy Code
 - ◎ deflates the mass - less file size
 - ◎ simple elegance - improve browsing times

CSS: Purpose

Created by

- W3C standard for defining the presentation of documents
 - HTML, XHTML & any XML language

Purpose

- Allows (X)HTML to work on document structure
- CSS w/ its own syntax works on presentation

CSS: Benefits



- ⦿ Overview of why to use CSS
 - ⦿ Better type and layout controls
 - ⦿ (X)HTML could do presentation, but it is not as accurate or robust in control over type, layouts & backgrounds
 - ⦿ Less work
 - ⦿ developing a couple of style sheet can control a 500 page website - (X)HTML presentation would have to be controlled on every page. Yikes!!

CSS: Benefits



- ⦿ Overview of why to use CSS
 - ⦿ Potential for smaller documents
 - ⦿ removing presentation from (X)HTML can reduce doc size (i.e. font element & nested tables)
 - ⦿ again ... one page in CSS controls many docs
 - ⦿ More accessible sites
 - ⦿ cleaner, more straight forward sites allow for better accessibility and more meaningful tags in the actual (X)HTML code
 - ⦿ style sheets written for different devices make browsing on PDA, phone, TV or printing a document a lot easier

CSS: Benefits



- Overview of why to use CSS

- Reliable browser support

- This is the standard - browsers are starting to better follow the standards set by the W3C :) (Y'OULD THINK THEY WOULD ... THEY ARE ON THE W3C ... GO FIGURE!)

CSS: Standards



◎ CSS Level 1 Recommendation

◎ CSS1 - released 1996

- ◎ properties for adding font, color and spacing instructions
- ◎ took a few years for browsers to catch on to a good thing and for CSS 1 to gain wide support

◎ CSS Level 2 Recommendation

◎ CSS2 - released 1998

- ◎ add positioning properties for page layout
- ◎ styles for different media
 - ◎ print, handheld
 - ◎ aural - blind and print-impaired
- ◎ more sophisticated methods for selecting styling elements

CSS: Standards

- ◎ CSS Level 2, Revision 1

- ◎ CSS2.1

- ◎ minor adjustments to CSS 2
 - ◎ latest working version

- ◎ CSS Level 3

- ◎ CSS3 - still in development

- ◎ some browsers offer support

- ◎ vertical text support!
 - ◎ improved table handling
 - ◎ improved international language support
 - ◎ better XML integration
 - ◎ larger list of color names (I hate hex numbers!)
 - ◎ multiple background images in a single element

CSS: How they work



- ⦿ Start with (X)HTML document w/ markup
- ⦿ Think about what you want it to look like
- ⦿ Connect style rules to the doc

CSS: Syntax



- ◎ CSS styles are specified in “rules”
 - ◎ rule are made up of “selectors”, “properties” & “values”
 - ◎ selector
 - ◎ specifies what HTML elements are to be affected by the style
 - ◎ property
 - ◎ specifies what we want to change about the element’s appearance
 - ◎ value
 - ◎ specifies what we want the proper to look like
 - ◎ can be color values, length measurements or other keywords

CSS: Writing the rules



⦿ single element selector / descendant selector

⦿ `h1 {color:#f00;}`

⦿ h1 - selector

⦿ color - property

⦿ red - value

⦿ all `<h1>` elements will be in red

⦿ `color:#f00` - is called a declaration

⦿ made up of a property/value pair

⦿ a colon separates the property from the value

⦿ provides the instructions

CSS: Writing the rules



example - multiple value pair

```
h1 {color: red; font-style:italic;}
```

- all `<h1>` elements will be in red and italicized
- `color:red;` - declaration block
- `font-style: italic;`
- each rule on one line
- semi-colon separates one value pair from another
 - include last “;” in a declaration block

CSS: Writing the rules



⦿ whitespace

- ⦿ minimize the whitespace (save character space)
- ⦿ no space between
 - ⦿ the opening curly brace of a declaration block and the first declaration
 - ⦿ a declaration's property and colon
 - ⦿ a declaration's colon and value
 - ⦿ a declaration's value and semicolon

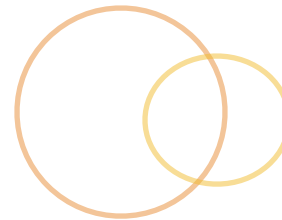
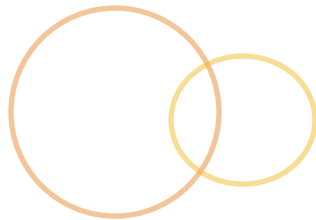
CSS: Writing the rules



- ① naming

- ① write styles in lower-case (just a suggestion)
- ① CSS is case-insensitive

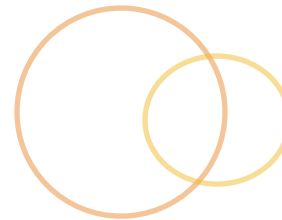
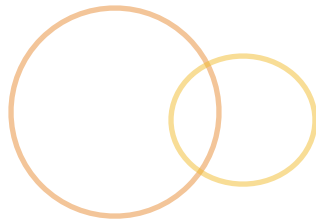
CSS: Help



resources

- Complete list of current properties in CSS2.1
 - W3C
 - www.w3.org/TR/CSS21/propidx.html
- CSS: The Definitive Guide, Second Edition
 - Eric Meyer
- Web Design in a Nutshell
 - Jennifer Robbins
- Web Standards Solutions: The Markup and Style Handbook
 - Dan Cederholm

CSS: Help



resources

- resources
 - The Zen of CSS Design: Visual Enlightenment for the Web
 - Dave Shea
 - Eric Meyer on CSS: Mastering the Language of Web Design
 - Eric Meyer
 - CSS: The Missing Manual
 - David Sawyer McFarland

CSS: Help

- resources - online

- A List Apart

- www.alistapart.com

- css-discuss

- www.css-discuss.org

- Stopdesign

- www.stopdesign.com

- Mezzoblue

- www.mezzoblue.com

- Meyerweb

- www.meyerweb.com

- Molly.com

- www.molly.com

- Simplebits

- www.simplebits.com

- Kamren's Slides

- www.kamren.biz

CSS: Selectors

multiple element selectors

```
h1, h2, h3 {color :#f00; font-style:italic;}  
p {font-size:1.1em;}
```

- ◉ `<h1>`, `<h2>` & `<h3>` elements will be red(`#f00`) and italic
 - ◉ group selectors
 - ◉ apply same style rule to a number of elements
 - ◉ separate selectors with a comma
- ◉ `<p>` elements have a font size 110% percent of the default text or 1.1x the default text size