

# **Cascading Style Sheets:**

## **Formatting & Positioning Using Standards**

**R. Scott Granneman**

**Senior Consultant, Internet Services**

**Bryan Consulting**

© 2003 Scott Granneman

Last modified 12 July 2003

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

# Our Topics

- Why CSS?
- Our CSS toolset
- Integrating styles with your HTML
- CSS programs & tools

# Why CSS?

- Separate content from presentation
- Apply different styles to same content in different media: Web browser, cell phone, auditory, print
- Adherence to standards

# CSS & Time

- Yes, CSS does take longer at the beginning
  - Takes longer to plan the Web site
  - Takes longer to set up the Web site
- However, CSS vastly reduces maintenance time
  - Alter 10,000 Web pages by making one change on one CSS page!
- Maintenance is the hidden part of the iceberg
  - Will take up most of your time
  - Most people never plan for it, however

# Basic CSS Syntax

- In HTML, the basic syntax is:  
`<element attribute="value">`
- In CSS, the basic syntax is:  
`selector {property:value;}`

- `selector {property:value;}`
  - selector: describes the HTML you wish to style
  - property: describes the thing you want to change in the HTML
  - value: describes the actual formatting you wish
- Example: `p {color:red;}`
  - selector = “p” = you wish to alter the `<p>` element
  - property = “color” = you want to change the color of text inside `<p> ... </p>`
  - value = “red” = you want the color to be red

# HTML & CSS Sample Code

## HTML

```
<p>  
  This is a paragraph  
</p>  
<blockquote>  
  A quotation: "Time flies like an  
    arrow; fruit flies like a banana."  
  End of quotation.  
</blockquote>  
<p>  
  Another paragraph.  
</p>
```

## CSS

```
p {font-family:"Times New  
  Roman",Times,serif; font-  
  size:14px;}  
blockquote {font-  
  family:Verdana,Arial,sans-serif;  
  font-size:12px;}
```

# The Results

This is a paragraph.

A quotation: "Time flies like an arrow; fruit flies like a banana." End of quotation.

Another paragraph.

# Our Toolset: Basic CSS

- HTML:

`<element attribute="value">`

- CSS:

`selector {property:value;}`

- Examples:

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

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

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

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

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

# Toolset: CSS Selectors

- Selector can be any one of the following:
  - HTML element
    - Affects every instance of an element
    - Example: `p {color:red;}`
  - Class
    - Affects multiple elements you associate with the class
    - Example: `.attention {color:red;}`
  - ID
    - Affects one unique element per page with the ID
    - Example: `#attention {color:red;}`

# HTML Elements as Selectors

- Every instance of the element is changed
- Example:  

```
p {color:red; font-weight:bold;}
```
- Results: every `<p> ... </p>` will be red & bold
- My rule: if you want more than 51% of all `<p>` elements to be red and bold, then go ahead and use `"p {color:red; font-weight:bold;}"`

# 1. Selector: HTML Element

- Used when you want to affect every instance of an element
- Example:

```
p {color:red; font-weight:bold; }
```
- Results: every `<p> ... </p>` will be red & bold
- Advantages: changes every instance of the selected HTML element
- Disadvantages: changes every instance of the selected HTML element
- Use this with caution!

# How I Use HTML Elements as Selectors

- In CSS:

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

- Within HTML:

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

- Results

- Every element in which text could be used on the entire site has a specific, defined font
- Unified look & design

- Note that I am using **selector grouping** with commas

## 2. Selector: Class

- Used when you want to apply a class to any appropriate element, as many times on a page as you desire
- Example (note the dot!):  

```
.attention {color:red; font-weight:bold;}
```
- HTML results (note the lack of a dot!):
  - o `<p class="attention"> ... </p>` is red & bold
  - o `<p> ... </p>` is not
  - o `<td class="attention"> ... </td>` is red & bold
  - o `<td> ... </td>` is not
  - o `<hr class="attention">` does nothing

# Class Names

- Rules
  - No spaces or underscores
- Bad class names:
  - .author name
  - .#1\_title
- Good class names:
  - .authorName
  - .title1
  - .footnote

# The Big Rule About Class Names

- Class names should describe **function**, not appearance!
- Bad class names:
  - .bold
  - .small
  - .bigRed: What if you later decide to define text as **green**, not **red**? Now your name doesn't match the appearance!
- Good class names:
  - .emphasis
  - .title
  - .footnote

# HTML Elements + Classes

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

- Do this in CSS:

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

- Do this in HTML

```
<p class="title">Madam, I'm Adam</p>
```

# 3. Selector: ID

- Used when you want to define a unique style for an appropriate element, once per page only
- Example (note the pound sign!):  

```
#navigation {color:#00FF00; font-size:14px; }
```
- HTML results (note the lack of a pound sign):
  - `<p id="navigation"> ... </p>` is green & 14px
  - `<p> ... </p>` is notOR ...
  - `<td id="navigation"> ... </td>` is green & 14px
  - `<td> ... </td>` is not
- Remember: only once per page!

# More Sophisticated Selectors

- Contextual Selectors
  - Example: `td p {font-size:10px;}`
  - Meaning
    - Any `<p>` inside a `<td>` will be 10px
    - Any other `<p>` is unaffected

# Toolset: `<span>` & `<div>`

- `<span>` & `<div>` were introduced as HTML elements designed solely as selectors for CSS
- All other HTML elements do something even if they do not have attributes & values
  - `<p>`, for instance, creates a paragraph
  - `<td>` creates a table cell
  - 2 exceptions: `<img>` & `<a>`
- `<span>` & `<div>` by themselves, however, do **nothing** on a Web page
  - They **MUST** have an CSS-based attribute/value pair (`class="x"` or `id="x"` or `style="x"`) to do anything productive

# Block-Level & Inline

- HTML elements in <body> are either block-level or inline
- Block-level
  - General purpose: describe structure of document
  - In the browser, block-level elements begin on a new line
  - Defines a large section of a document
  - Can contain inline & block-level elements
  - Ex's: blockquote, h1, li, ol, p, pre, table, td, ul
- Inline
  - General purpose: describe appearance of document
  - Can contain inline elements; CANNOT contain block-level elements
  - Defines a smaller section inside another section
  - Ex's: a, b, br, font, i, img

# <span>

- Inline HTML element
- <span> ... </span> does nothing
- Use <span> to hold attribute/value pairs relevant to CSS:

```
<p>
```

```
    Scott is the <span  
    class="emphasis">man</span>.
```

```
</p>
```

# <div>

- Block-level HTML element
- <div> ... </div> does nothing
- Use <div> to hold attribute/value pairs relevant to CSS:

```
<div class="navigation">  
  <p class="off"><a  
href="index.htm">Home</a></p>  
  <p class="on"><a  
href="contact.htm">Contact</a></p>  
  <p class="off"><a  
href="widget.htm">Widgets</a></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 very <span class="emphasis">  
    important</span> that you understand  
    the difference between span and div,  
    or you are <span class="emphasis">  
    doomed</span>!  
  </p>  
</div>
```

# Key Concept: Inheritance

- Let's say you've got this CSS:

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

- And you've got this HTML (don't worry about `<span>` for now):

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

- "Seamus" is going to be Times New Roman, 12px, & bold – why?

# Inheritance Pondered

- 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

# Integrating Styles with HTML

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

# 1. Inline Styles

- **Before:**

```
<p>  
  <font face="Times New Roman, Times, serif"  
    size="12" color="red">Scott is your  
  man.</font>  
</p>
```

- **After, using the `style` attribute:**

```
<p style="font-family:'Times New Roman', Times,  
  serif; font-size:12px; color:red;">  
  Scott is your man.  
</p>
```

# Inline Styles: Good & Bad

- Quick & easy to create, but difficult & time-consuming to manage
- Must repeat over and over for every element (if you want every `<p>` on a page to be red, bold, & Times New Roman ...)
- Can't change the style according to the media, so styles apply to all media
- Doesn't really separate content & presentation
- Use inline styles for unique instances

# 2. Embedded Styles

- Place `<style> ... </style>` in `<head>`
- Example:

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

# Notes on Embedded Styles

- Embedded styles are great for one page ...
- ... but they rapidly become difficult to manage on multiple pages

# 3. Linking to External Styles

- 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="address/to/style/sheet.css">  
</head>
```

# A Note on the Path to Your CSS File

- Path to CSS file can be any of the following:
  - href="client.css"
    - Relative path
    - Useful only if all your files are in the same directory, which is silly
  - href="http://www.client.com/css/client.css"
    - Absolute path
    - All the disadvantages of absolute paths, so avoid
  - href="/css/client.css"
    - Absolute path with advantages of relative path
    - I **always** place a css directory in my root directory, and I **never** move it, so recommended

# External Style Sheet

- Example, saved as "smith.css":

```
/* CSS for Smith Inc.; Created 27 Sept. 2002 */
```

```
/* Common */
```

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

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

```
/* Navigation */
```

```
.on {font-size:14px; font-weight:bold;}
```

```
.off {font-size:14px;}
```

# CSS Comments

- Use comments in CSS for the same reason you use them in HTML
  - Notes to yourself & others
  - Debug: comment out troublesome CSS for testing

- **HTML comments:**

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

- **CSS comments:**

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

# Thoughts on Linking to External Styles

- If you've got more than one page, you really ought to use an external style sheet
- You can now change the look & behavior of a site by changing only one document!

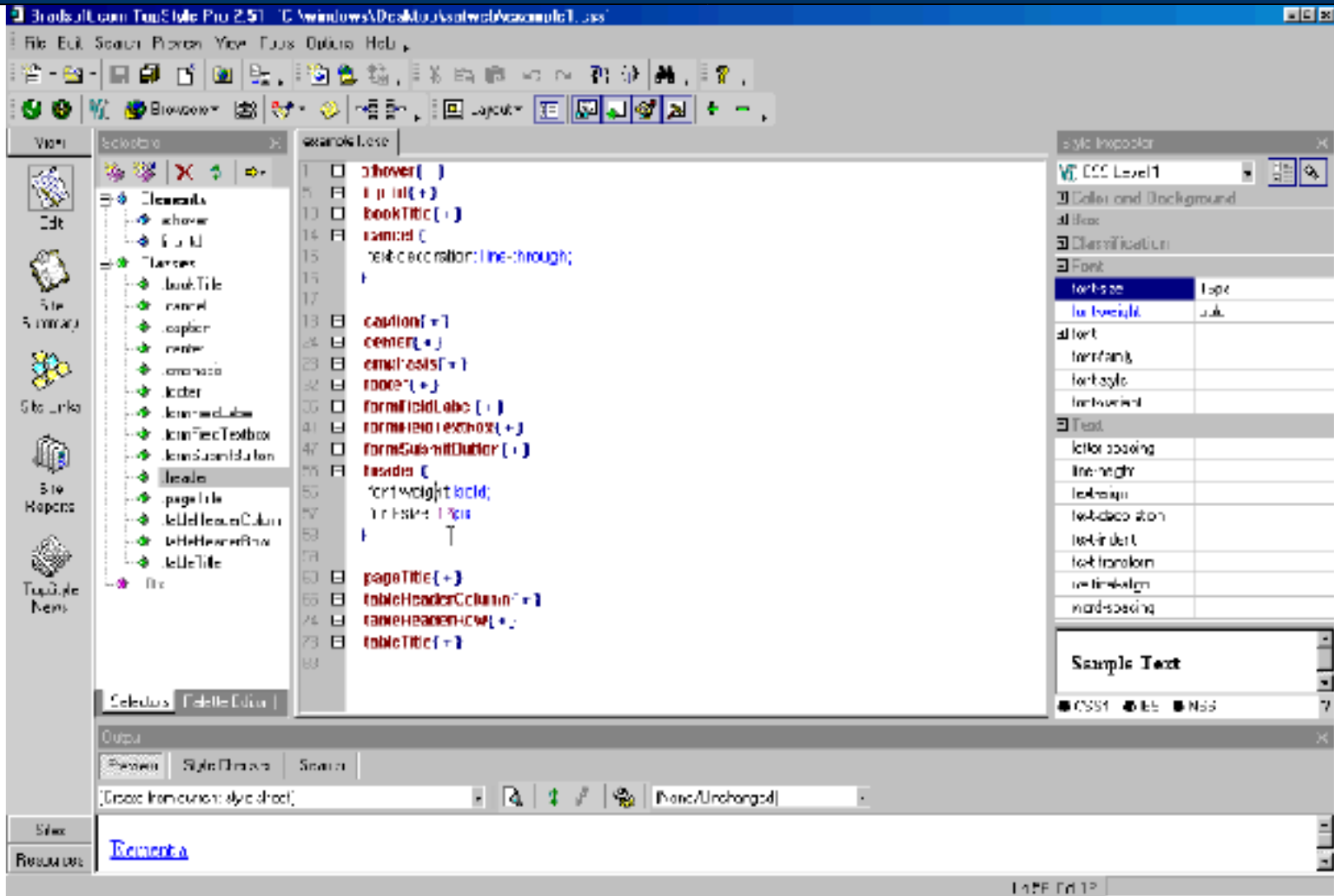
# CSS Programs & Tools

- CSS editors
- Browser tools
- Web sites
- Books

# CSS Editors

- TopStyle ~ [www.bradsoft.com](http://www.bradsoft.com)
  - Windows only
  - Lite (free) & Pro (\$80)
  - Highly recommended
- Style Master ~ [www.westciv.com](http://www.westciv.com)
  - Windows & Mac
  - \$49
  - Some problems, but excellent help & tutorials
- Morphon CSS Editor ~ [www.morphon.com](http://www.morphon.com)
  - Java-based, so Windows, Mac, & Linux
  - \$39
  - Works well, especially on a fast machine

# TopStyle Pro



# Browser Tools

- Mozilla Sidebar
- Bookmarklets
- W3C CSS Validation

# Mozilla Sidebar

- Get sidebars from [www.granneman.com/webdev/browsers/mozillan/sidebars](http://www.granneman.com/webdev/browsers/mozillan/sidebars)
- Install sidebar
- Open sidebar
- Voila! CSS1 & CSS2 references available at your fingertips

# Mozilla CSS2 Sidebar

The screenshot shows a Mozilla browser window with the address bar displaying `http://www.w3.org/TR/CSS2/Reference/Defaul/prop-font-size.html`. The browser's sidebar is open to the 'CSS 2' section, with 'font-size' selected. The main content area displays the following information:

[Go to Standard](#)

**Property:** font size

**Inherited:** yes, the computed value is inherited

**Value:** <absolute-size> | <relative-size> | <length> | <percent

**Initial:** medium

**Applies to:** all elements

**Media:** visual

**Percentages:** refer to parent element's font size

**Examples:**

<pre>&lt;AAA&gt; AAA   &lt;BBB&gt;BBB&lt;/BBB&gt;   &lt;CCC&gt;CCC&lt;/CCC&gt;   &lt;DDD&gt;DDD&lt;/DDD&gt;   &lt;EEE&gt;EEE&lt;/EEE&gt;</pre>	<p><a href="#">View</a></p> <pre>* {display: block} AAA {font-size: 10pt} BBB {font-size: 150%} CCC {font-size: 8pt} DDD {font-size: 1.5em} EEE {font-size: xx-large}</pre>
------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------

# Bookmarklets

- Tiny JavaScripts, saved as bookmarks, that act like simple programs
- Some of my favorites
  - Resize browser window to 800x600 or 1024x768
  - Remove background image
  - Number of links
  - Validate HTML

# Acquiring Bookmarklets

- Find bookmarklets
  - [www.bookmarklets.com](http://www.bookmarklets.com)
  - [www.squarefree.com/bookmarklets/](http://www.squarefree.com/bookmarklets/)
  - [www23.brinkster.com/favelets/favelets.htm](http://www23.brinkster.com/favelets/favelets.htm)
- Right-click on bookmarklet & choose "Bookmark This Link"
- My favorite CSS bookmarklets
  - View style sheets
  - Test styles
  - Disable stylesheets
  - W3C CSS validator

# Test Styles Bookmarklet, Before

## How a Bank Got E-Mail Scammed

Ann Marie Poet's new business partner called himself Dr. Mbusc Nelson, and said he was an AT&T.

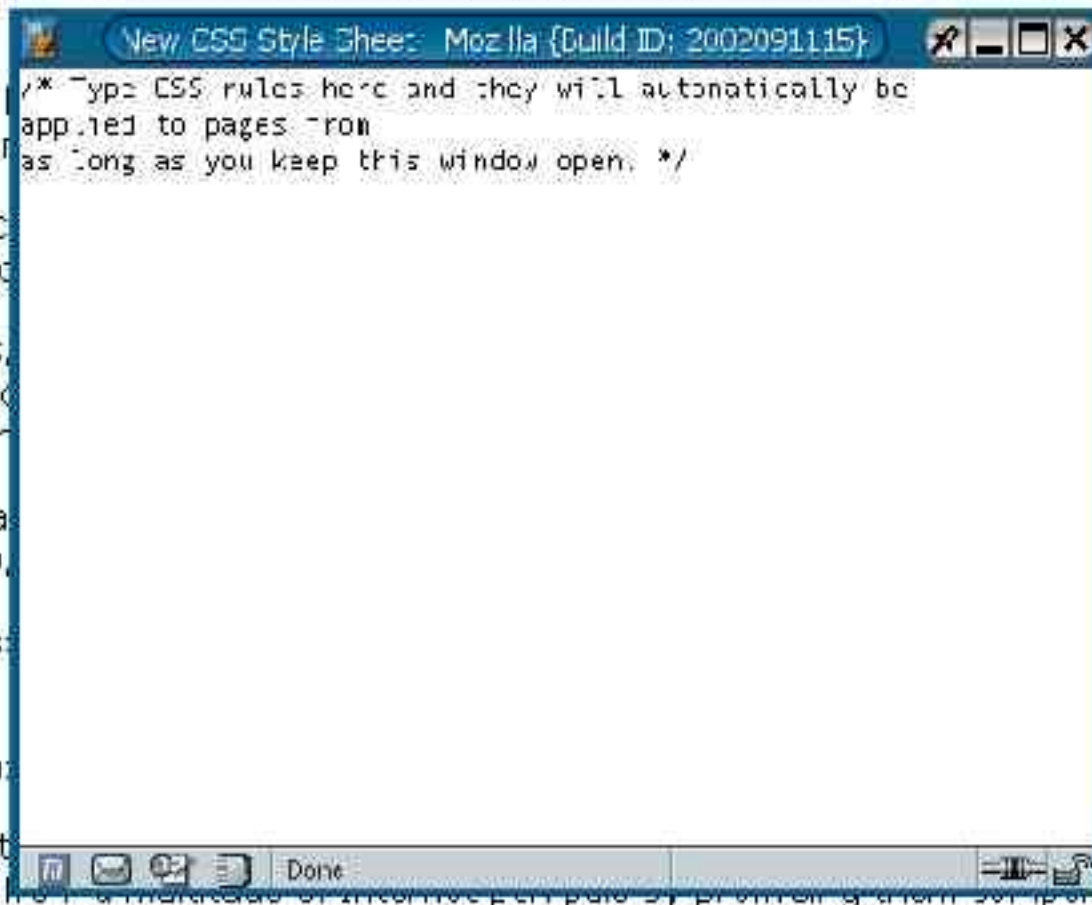
Nelson popped into Poet's e-mail box and transferred \$1.8 million from her account to his. This struck Poet as a reasonable amount of money that could be transferred.

According to court records, Nelson transferred the money from the accounts in South Africa and Poet was indicted last Friday and is serving 18 months in prison and a \$250,000 fine.

The embezzlement was discovered by a bank teller who reported it to insufficient funds.

Poet reportedly believed that Nelson was a doctor and a successful businessman.

The scam Poet fell for is a type of scam where a person successfully extorts money from a victim by promising to provide a service or product that is never delivered.



# Test Styles Bookmarklet, After

## How a Bank Got E-Mail Scammed

Anni Marie Poel's new business partner called himself Dr. Mbuso Nelson, and said

Nelson popped into Poel's e-mail account and said he was a doctor who needed assistance.

This struck Poel as a red flag, but she transferred the money before the money company was exposed.

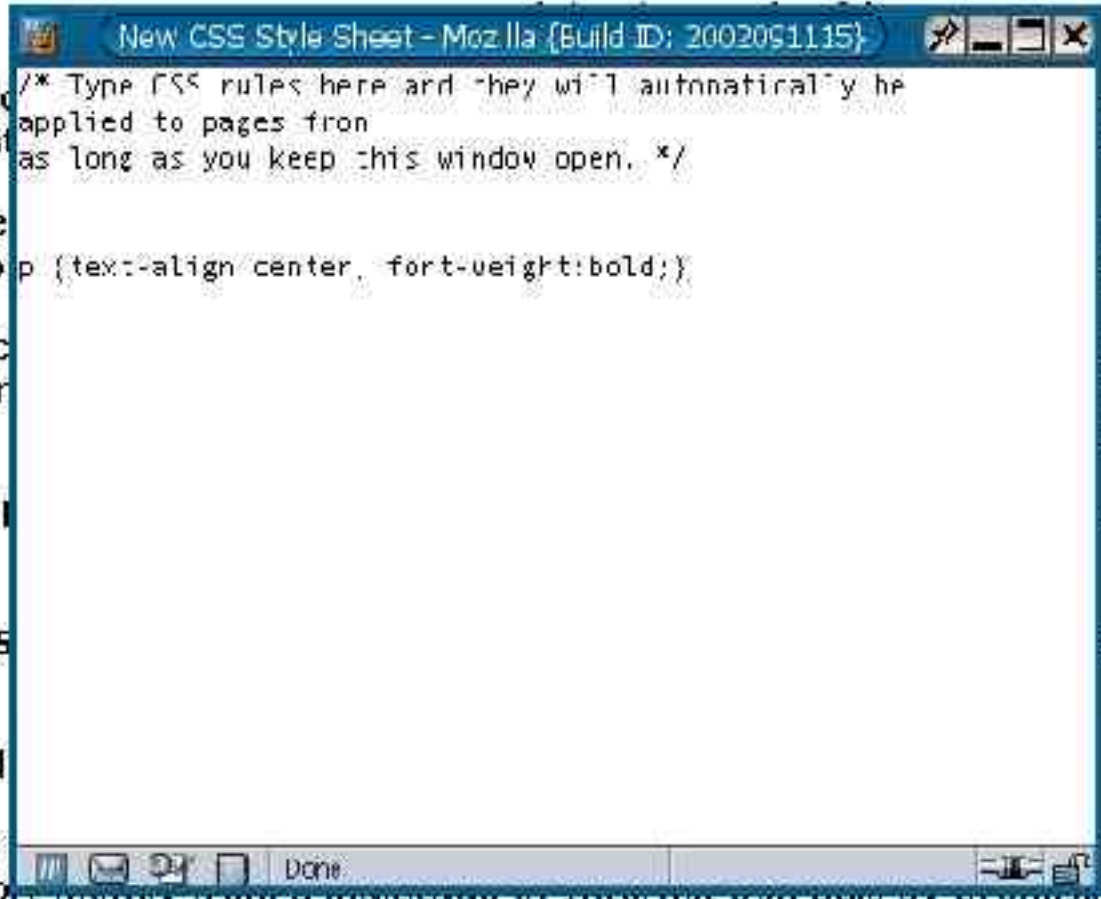
According to court records, Poel transferred \$360,000 per wire transfer.

Poel was indicted last month for wire fraud.

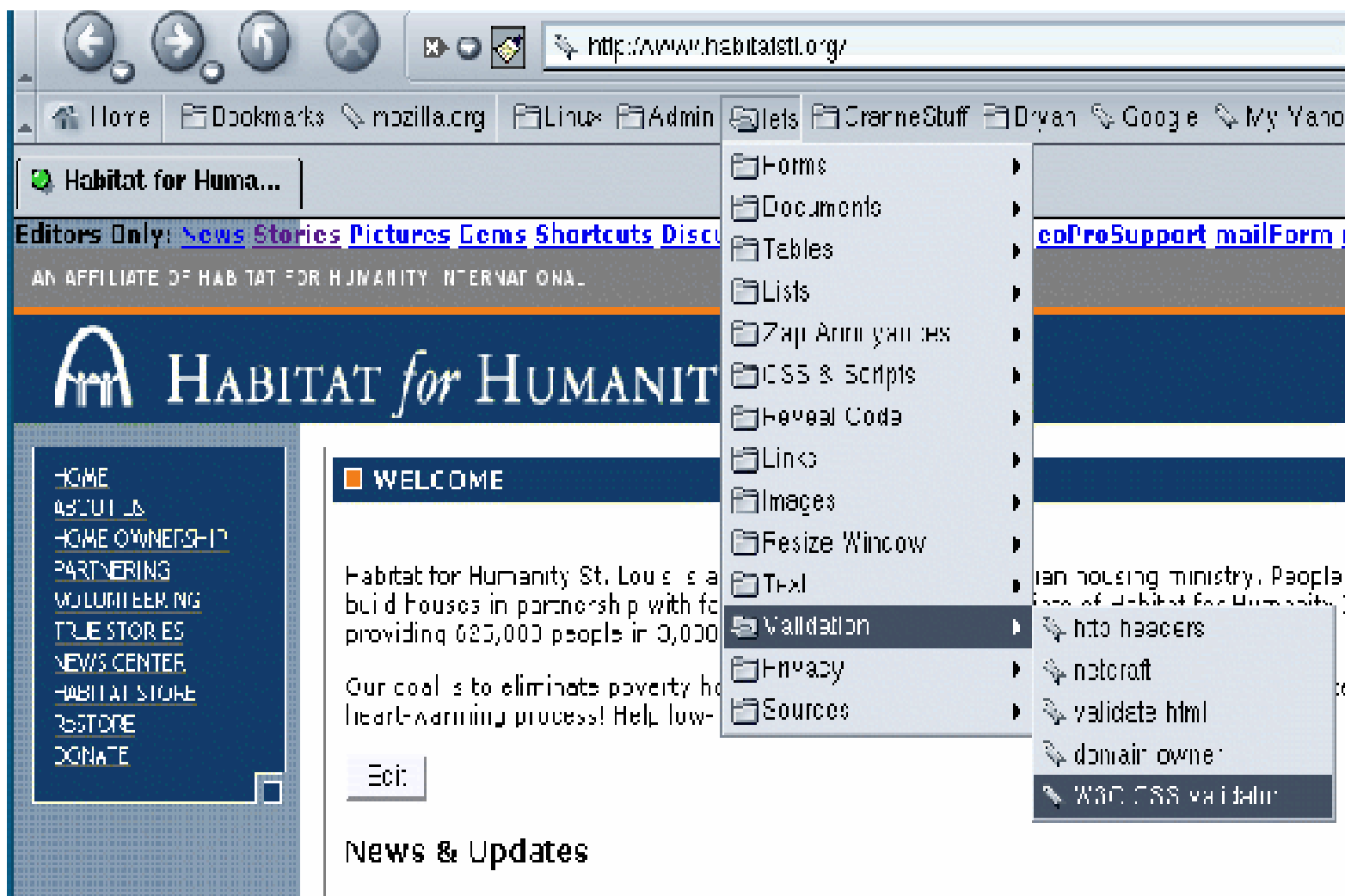
The embezzlement was discovered by a bank employee.

Poel reported the fraud to the FBI.

The scam Poel fell for is a common one, where a scammer impersonates a doctor who successfully extorts money from a miscreant or someone who is not a doctor.



# W3C's CSS Validator Bookmarklet



# W3C's CSS Validator Bookmarklet, Post-Submission

## W3C CSS Validator Results for [http://www.ha](http://www.habitatstl.org/)

To work as intended, your CSS style sheet needs a correct document parse via [Valid HTML](#)

### Errors

URI : <http://www.habitatstl.org/>

- Line: 40 Context : .container  
Invalid number : [padding:none](#) is not a padding-top value : none
- Line: 42 Context : .leftnavtable  
Invalid number : [padding-top:-5.0](#) negative values are not allowed : -5j

### Warnings

URI : <http://www.habitatstl.org/>

- Line : 91 Level : 1 You have no color with your background-color : body

# Web Sites

- Annotated collection of great CSS links at [www.granneman.com/usefulli/developm/coding/css/](http://www.granneman.com/usefulli/developm/coding/css/)

# Books

- Eric A. Meyer
  - *Cascading Style Sheets: The Definitive Guide*
    - Detailed overview of CSS 1
  - *Cascading Style Sheets 2.0 Programmer's Reference*
    - Quick reference guide, with focus on CSS 2
  - *Eric Meyer on CSS: Mastering the Language of Web Design*
    - Project-focused: details 13 projects, showing how to use CSS and explaining why particular methods were chosen



# Thank You!

- Final thought: CSS is more difficult, but it gives you greater standards-based control over your Web pages.
- Email me: [scott@granneman.com](mailto:scott@granneman.com)
- Visit my Web site: [www.granneman.com](http://www.granneman.com)
- Join GranneNotes, my newsletter for people interested in technology & the Internet, by visiting [www.granneman.com](http://www.granneman.com)

# 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](mailto:scott@granneman.com)