Cascading Style Sheets

Dave Edsall

IAGenWeb
County Coordinator's Conference
June 30, 2007

What We Will Cover

- Enough to get you started
- A little history for context
- Why we want to use CSS
- Basic rules used by all styles
- Exercises to practice what you will have learned

What We Will *Not* Cover

- Every detail of CSS – that's homework for you
- How the *cascade* works
- How inheritance works
- Margins, borders, padding and how the box model works. (*This isn't hard to learn*)
- Positioning

So, Let’s Get Started!

Before CSS - Style over Substance

- Everyone wanted *cool* web sites
- Lack of standards enabled browser wars
- Netscape and Microsoft *polluted* HTML with their own tags
  - Remember <blink>?
  - Do you use <font>?
  - What about <b> or <i>?
  - What about tables for formatting? Huh? You ARE doing that, aren't you?

Stop! - Standards

- Some thought this was crazy
- W3C steps in and develops standards for styling web sites
- HTML standardized with styles removed (4.0)
- CSS standardized (currently 2.1, 3.0 in the works!)

CSS Permits Style *and* Substance

- Styles are separated from content:
  - HTML holds the content and structures it (paragraphs, sections, pages, lists)
  - Style sheets tell the browser how to *present* the content.
- You can be the author and the typesetter and graphic designer
- Or, you can be the author and let someone else handle the look-and-feel

How to Add Style

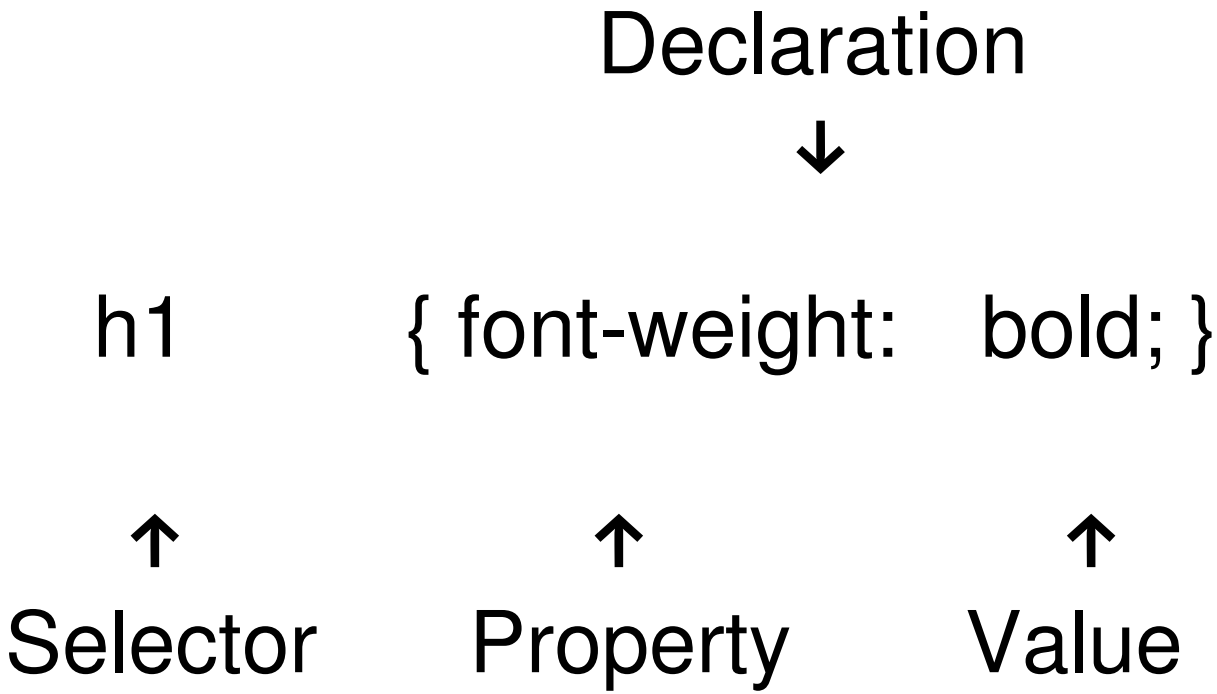
- Include it inline – put it right into your tags with **style="…"** attribute
  - Useful for *one-offs*
- Include it in the document, using **<style>** and **</style>**
  - Useful for single pages
- Import it from an external file or location with **@import** or **link**
  - Really useful for styling a complete web site

What Can I Style?

- HTML and XML elements like **body**, **p**, **a**, **div**, **span**
- Element *classes*
- Particular elements labelled with *ID*s

CSS Rules

- A CSS rule looks like this



Selectors - Elements

- Let’s give a paragraph a background color

  <p style=“background-color: yellow;”>

- Note – there is no selector. The element is its own selector

Exercise 1 – Inline style

- Start Notepad
  - Click **start** -> All Programs -> Accessories -> Notepad
- Open the file exercise1.html in the folder css
  - Click File -> Open
- Look for the paragraph that says “This should be text on a yellow background”
- In the **p** tag, add the style attribute

  style="background-color: yellow;“
- Save the file
- Open the file with your browser. Is the background yellow?

Selectors - Elements

- Even cooler! Let's give all paragraphs a *border* colored red

```
<style>
p {
        border-width: 2px;
        border-style: solid;
        border-color: red;
  }
</style>
```

- ***Note*** *– we can apply multiple declarations to an element*

Exercise 2 – Document Style

- Start Notepad
- Open the file exercise2.html in the folder css
- Inside the <head> container, after the <title>, add the following:

```
<style>
  p {
    border-width: 2px;
    border-style: solid;
    border-color: red;
  }
</style>
```

- Save the file
- Open the file with your browser. Are the borders all red?

Selectors - Classes

- Does it have to be one or all?
- What if I want some styles for some paragraphs and other styles for other paragraphs?
- Define *classes* of styles
- Especially useful for designing entire web sites.

Selectors - Classes

Classes are defined by using a period (.) in the selector

```
p.intro {
   font-size: large;
   font-style: italic;
}

p.obit {
   font-size: small;
   font-family: sans-serif
}
```

Selectors - Classes

Classes are used with the **class** attribute in your elements

Herein shall ye find the final epitaths of the Lindsay clan who did so bravely defend Castle Edzell

David John Lindsay didst fall in battle after consuming an excessive amount of English cheese.

Selectors - Classes

- Classes are not defined inline.
- Class can be defined in the document
- Classes are most often defined in external style sheet files that are imported into several documents

Exercise 3 – Classes

- Start Notepad
- Open the file exercise3.html in the folder css
- Inside the <head> container, after the <title>, add the following:

```
<style>
p.intro {
   font-size: x-large;
   font-style: italic;
}

p.obit {
   font-size: xx-small;
   font-family: sans-serif;
}
</style>
```

Exercise 3 – Classes (cont.)

- Look for the paragraph with the text “Edsall Family Obituaries
- In the **p** tag, add the **class** attribute
  class="intro“
- Look for the paragraph with the text Delbert Gary Edsall
- In the **p** tag, add the **class** attribute
  class=“obit“
- Save the file
- Open the file with your browser. Do things look as you expect?

Selectors - IDs

- What about a style for the elements that occur only once in every document?
- Every element in a document is an *object* in the *Document Object Model* (**DOM**)
- Every object in the DOM can have an ID.
- Associate a style with a given ID

Selectors - IDs

- IDs are defined using an octothorpe (**#**)

```
div#navbar {
   background-color: tan;
   margin-left: 15px;
   margin-right: 5px;
   border: solid black 1px;
}
```

Selectors - IDs

IDs are used with the **id** attribute in your elements (surprise, surprise)

```
<div id="navbar">
   <ul>
      <li><a class="selected" href="…
      .
      .
   </ul>
</div>
```

Selectors - IDs

- Like classes, IDs are not defined inline, can be defined in the document and are most often defined in external style sheet files that are imported into several documents

- An ID should be unique to each element. But browsers will let you get away with using IDs on more than one element

Real Cool! – Psuedo Classes

- Called *psuedo* because they apply to *events*
- Events aren't static objects in your document
- Event happens – style is applied
- Most often associated with mouse actions

Psuedo Classes – Mouse Events

Example - Links

- Before the mouse gets there – **link**
- Mouse over – **hover**
- Mouse click – **active**
- After the mouse leaves - **visited**

Psuedo Classes – Making a *Rollover*

- Psuedo classes are defined using an colon after the element (:)

```
div#navbar a {
   color: #fbf093;
}

div#navbar a: hover {
   color: #000;
   font-weight: bold;
}
```

Psuedo Classes – Making a *Rollover*

- The browser then determines when the style needs to be applied depending on what the user does with their mouse

```
<div id="navbar">
  <ul>
    <li><a href="…
    <li><a href="…
    .
    .
  </ul>
</div>
```

Multiple Selectors

- Wait! What's going on with **div#navbar a**?

```
div#navbar a {
   color: #fbf093;
}
```

- Just as a selector can have multiple declarations, a declaration block can be associated with multiple selectors.

- Here we require that the anchor (**a**) tag be associated with a div having **id=navbar** or the style will not be applied. This is POWERFUL!

Who Put the Cascade in CSS?

- When multiple selectors and styles are associated with an element, what wins?

```
a: link { color: blue; }

div#navbar a { color: black; }
```

- The browser *cascades* from lesser importance to higher importance. Whichever declaration is more specific wins. In this case, if we are in a div with id=navbar, the style we have defined for **a** tags in these divs wins.

Let’s Build a Site!

Exercise 4 – External Style Sheets

- In the folder CSS you will find an external style sheet named style.css. Open it with Notepad and study it if you wish.
- We are going to use tables to layout the page.
- We will have a title bar, a navigation bar and a content area
- We want the links in our navbar to roll over

Exercise 4 – Importing the Style Sheet

- Open the file exercise4.html

- Inside the <head> container, after the <title>, add the following:

```
<style type="text/css">
@import url(style.css);
</style>
```

Exercise 4 – Creating a Title Bar

After the **<body>** tag, add

```
<table class="container">
  <tr>
     <td class="titlebar" colspan=2>
     <h1>Pocahontas County</h1>
     </td>
  </tr>
```

Exercise 4 – Creating a Navigation Bar

- Then add

```
<tr>
<td class="nav">
<div id="navbar">
<ul>
```

- For each link to another page, add something like

```
<li><a href="records">Records</a>
```

Exercise 4 – Adding Content

- Then add

```
  </ul>
</div>
</td>
<td class="content">
```

to close off your navigation bar and begin your content

- Then type in any content you wish

Exercise 4 – Finishing Up

- Close off the content and the table by adding

```
 </td>
  </tr>
</table>
```

- Save the file
- Open your new page with your browser. Do you have a titlebar at the top? Do you have a navigation bar on the left? Do you have content on the right?
- You could even add a footer to your page by adding another table row at the end.

You Want Power?

- You can use this page as a *template* for all your pages
- If you later decide to change the color of the links on all the pages, you only need to change ONE file - **style.css**
- You have the power to make changes to a large site by modifying one file.

Questions?