



PIC 40A

Lecture 5: Introduction to Cascading Style Sheets (CSS)

What is CSS?

CSS

- Is a way to add style to XHTML documents
- Consists of rules for specifying how an element should be presented

The latest W3C standard is CSS2
(CSS3 coming soon)

Why CSS?

CSS

- keeps **presentation layer** separate from **structure layer** of Web design
- makes Web docs smaller, reduces clutter
- can be interchanged without major change to the XHTML document
- can impose a uniform style to a variety of XHTML documents with one style sheet
- can format different elements with a single style rule
- can format elements of the same type differently

Why the name “Cascading”?

Style sheets can be defined at 3 levels:

1.inline (low-level: most specific)

2.document

3.external (high level: most general)

More specific rule overrides more general rule

- Inline-applies to single element (deprecated in XHTML 1.1)
- Document-applies to whole XHTML document body
- External-applies to a collection of XML documents (CSS stored in a separate .css file)

Another reason is inheritance, which we will discuss later.

CSS Syntax

- CSS is a list of style rules placed as either
 - the content of an XHTML style element
 - a `.css` file (external level)
- each style sheet rule looks like:

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

A **selector** is an expression that specifies which elements to style. The expression

```
{property:value; ... }
```

is called the rule's **declaration**.

A Style Rule Example

```
p {  
  color: blue;  
  font-size: 11pt;  
  font-weight: bold;  
}
```

p is the style rule **selector**

color, font-size, font-weight are **properties**

blue, 11pt, bold are their corresponding **values**

The style rule **declaration** is the following:

```
{  
  color: blue;  
  font-size: 11pt;  
  font-weight: bold;  
}
```

Using Style Sheets in XHTML

- Inline level:

```
<p style="background: blue;  
color:white;">Paragraph with blue  
background and white text</p>
```

Inline style is **deprecated**

We will not be using inline style

Using Style Sheets in XHTML

1.

- Document level

```
<style type="text/css">
  p {color: blue;}
  h2 { font-size: 18pt; }
</style>
```

the `type` attribute is required

the `style` element must be in the XHTML head element

Using Style Sheets in XHTML

- External level

Use the `link` element inside `head` element:

```
<link href="pic40a-css.css"
      rel="stylesheet"
      type="text/css"
/>
```

Simple Selector Forms

- Single element name

- Property applies to all occurrences of that element

```
p {color: blue;}
```

- List of element names

- Property applies to all occurrences of all the elements in the list

```
p, h2 { font-size: 18pt; }
```

- Asterisk (*)

- Property applies to all elements in the doc

```
* {color: blue;}
```

Simple selector forms continued

- Contextual selectors
 - Property applies to selectors with a certain family tree

Example:

```
ul li ol{ list-style-type: decimal; }
```

applies to any ordered list having a list item ancestor that has a unordered list ancestor

ul	li	ol
ancestor of li	ancestor of ol	selected element

Class selectors

- Allow elements of the same kind to have different styles

Example:

CSS:

```
p.bluetext{ color: blue; }  
p.redtext{ color: red; }
```

XHTML:

```
<p class="bluetext">I'll be blue</p>  
<p class="redtext">I'll be red</p>
```

Class selectors continued

- It is possible to have different kinds of elements to have the same style

Example:

CSS:

```
.important{ font-style: italic;}
```

XHTML:

```
<h1 class="important">Warning!</h1>  
<p class="important">Look Out!</p>
```

Id selectors

- Allows one specific element with a given id value to have a particular style:

```
#myparagraph{ font-weight: bold;}
```

```
<p id="myparagraph">This bold.</p>
```

```
<p>Not bold.</p>
```

Pseudo Classes

- Styles that apply when something happens to an element

```
a:link{ color: red; }  
a:visited{ color: purple}  
a:hover{color: blue; }  
a:active{color: yellow; }
```


LVHA (Love, HA!)

1. `a:link` (for links not yet visited)
2. `a:visited` (for links visited)
3. `a:hover` (applies when mouse is over link)
4. `a:active` (applies during act of clicking)

Order matters.

Putting `:link` or `:visited` last
would override `:hover` or
`:active` styles

Selector Example

See selector example for examples of all the concepts.

CSS inheritance

Child elements can inherit the properties of the parent elements

- Most properties can be inherited: font, text-alignment etc.

- Things like background image and padding are not inherited.

CSS inheritance

More specific definitions override earlier definitions

```
body {color: red;}
```

```
p{color: yellow;} /* Overrides body */
```

Child element may override a property of a parent element

Child element may add additional properties to the ones inherited from the parent.

-Grandchildren will then have the properties inherited from the original parent elements and also from their direct parents.

This is another reason why it is called cascading style sheets.

See example

CSS: What rule applies

List from general to specific:

Any rule that applies to all elements.

```
body {color: blue;}
```

Any rule that applies to all elements of a given type.

```
p {color: blue;}
```

Contextual selectors

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

Any rule that applies to a given class

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

Any rule that applies to all elements of a given type and in a given class.

```
p.important {color: blue; font-weight: bold;}
```

Any rule that applies to a specific id.

```
#title {font-size: 28pt; color: blue;}
```

CSS: What rule applies

Later rule in the style sheet overrides an earlier rule.

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

```
p {color:blue;}◀ This one wins because  
it is the latest rule!
```

Example

```
<head>
  <title>CSS example</title>
  <style type="text/css">
    #someid { background-color:#CCC; border:1px solid yellow; }
    div { border:1px solid blue; }
    .class1 { color:#fff; }
    .class2 { font-weight:bold; }
    a { font-style:italic; }
    a:link { text-decoration:underline; }
  </style>
</head>
<body>
  <div id="someid" class="class1 class2">
    <a href="#">
      This text inherits all the styles above - each a different way...
      Why is the border yellow?
    </a>
  </div>
</body>
</html>
```


Validate your CSS

The validator is at

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

Link is also on our homepage under helpful links section