

INTRODUCTION TO CSS

INTRO TO CSS

○ Covered in this lesson:

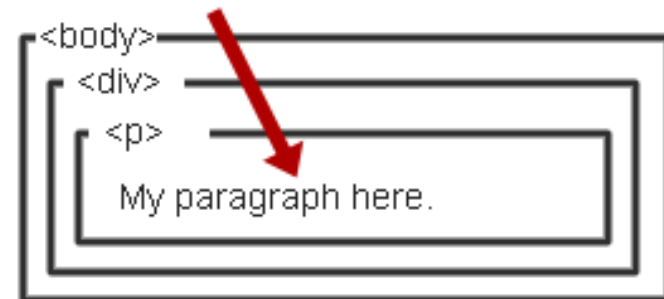
- Overview
 - What is CSS?
 - Why to use CSS?
 - CSS for Skinning your Website
- Structure
 - CSS Syntax Introduction
 - Three places CSS can be defined
 - CSS Syntax Specifics
 - Cascading Inheritance



WHAT IS CSS?

○ Cascading Style Sheet

- Stylesheet Language
 - Standards-based set of properties and attributes to define styles
- To describe the presentation a document written in a 'markup language' like HTML or XML
 - Markup encoding: `<p>My paragraph here.</p>`
 - Defines the style of how things in `<p>` tags appear.
 - Font, color, size, margins, etc.
- Cascading
 - Rules to determine how to
 - apply markup that contains
 - other markup



WHY CSS?

○ Separate Content

- Content is the text and images, marked up to define regions of specific types
- Form defines the “style” for the content

The old way:

```
<font size="14px">  
My First Header  
</font>  
<font size="12px" color="red" face="Verdana">  
My information 1 goes here.  
</font>  
<font size="14px">  
My Second Header  
</font>  
<font size="12px" color="red" face="Verdana">  
Different information goes here.  
</font>
```



WHY CSS? CONTINUED.

○ Separate Content

- Content

```
<p class="header">My First Header</p>  
<p class="info">My Information 1 goes here</p>  
<p class="header">My Second Header</p>  
<p class="info">Different Information goes here</p>
```

(Specific markup properties like Class will be discussed later).

- Style

```
.header { font-size:14px;}  
.info {   font-family: verdana;  
         font-color: blue;  
         font-size: 12px; }
```



WHAT DOES THIS SEPARATION GET US?

○ Separate Content

- Specify the style **once** for every instance of that class.
 - Example: Specify the font once for all text on the HTML page that you've identified as a "header".
- The stylesheet can be a **separate file** which all HTML pages on your entire site can link to.
 - Only have to specify the style once for your **ENTIRE SITE**
- Can change the style for your entire site by editing only ONE FILE.



CSS SKINNING

- “Skinning” - changing the look of a page or your site
 - Selecting an appearance by choosing which stylesheet to use.

```
<link rel="stylesheet" type="text/css" href="skin1.css" />
```

```
<p class="info">My Information 1 goes here</p>
```

+

```
skin1.css  
.info { background-color: White;  
        font-family: Verdana;  
        font-color: Blue; }
```

=

My Information 1 goes here.



CSS SKINNING 2

- “Skinning” - changing the look of a page or your site
 - Selecting an appearance by choosing which stylesheet to use.

```
<link rel="stylesheet" type="text/css" href="skin2.css" />
```

```
<p class="info">My Information 1 goes here</p>
```

+

```
skin1.css
.info {  background-color: Blue;
         font-family: Serif;
         font-color: White; }
```

=

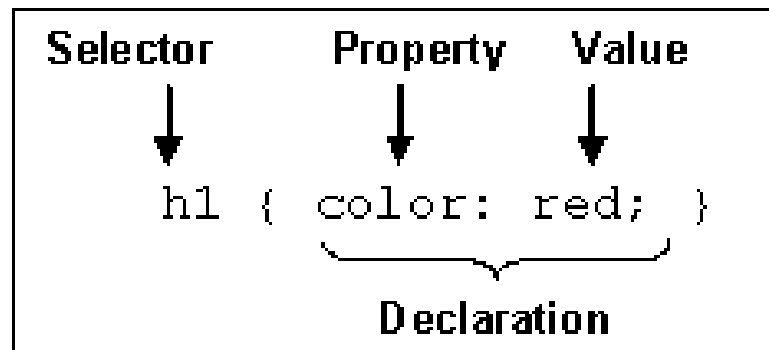
My Information 1 goes here.



CSS SYNTAX

○ 3 Elements to a CSS Statement

- Selector
 - What HTML sections does it affect?
- Property
 - What attribute of that HTML section will be affected?
- Value
 - What change will be made to that attribute?



THREE CSS DEFINITION LOCATIONS

- Inline: the “style” attribute

```
<p style="font-color:red;font-size:10px;">Content</p>
```

Note, the selector for inline CSS is the tag which contains the style attribute.

- Internal: the <style> markup tag

```
<html><head><style>
p {      background-color: Red;
          font-family: serif;
          font-color: White; }
</style></head><body>
<p>Content</p>
</body></html>
```

- External: the .css stylesheet file

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



CSS SYNTAX: SELECTORS

- There are many kinds of selectors and many ways to reference them:
 - Type, Class, ID, Pseudo, etc.

- HTML Type Tag – selected with the tag type

```
p {      font-size: 10px;  
        font-color: White; }
```

```
<p>Content</p>
```

- The Class Attribute – precede the class with a period

```
.myinfo { font-size: 10px;  
         font-color: White; }
```

```
<p class="myinfo">Content</p>
```

```
<div class="myinfo">Other content</div>
```



CASCADING INHERITANCE

- Nested elements inherit the properties from the its parent
- If you specify a style for the `<body>` tag it will affect all content in your HTML page.
- If you want to override inherited settings, you need to specify a style in a more local element



```
body {  font-family: Verdana;
        font-size: 14px; }
```

```
body {  font-family: Verdana;
        font-size: 1.1em; }
.littletext { font-size: 8px; }
```

```
<body>
This text is larger.
<p class="littletext">This text is
smaller.</p>
```