

**CSS**



Cascading  
Style  
Sheets

**SOEN 287 WEB PROGRAMMING**



# CSS TOPICS

- **What is CSS?**
- Levels of CSS
- Selectors
- Property Values
  - Fonts/Colors
  - Links
  - Lists
  - Alignment of text
  - Box Model
  - `<span>` vs `<div>`
- Bootstrap



# CASCADING STYLE SHEET (CSS)

- Style language to describe look and format of a document
- Allows for separation of document content from document presentation
- Can specify presentation styles and associate them with markup elements in webpage
- HTML
  - Was not intended to contain tags for formatting
  - Was designed to define content of a document (h1, p, ...)



# CASCADING STYLE SHEET (CSS) ...

- HTML 3.2 - tags like <font>, and color attributes were added (nightmare for web developers)
- HTML 4.0 - creation of CSS, so all formatting could be removed from the HTML document & stored in a separate file (.css)
- Can change the appearance and layout of all pages in a Web site by editing one file.

**Webpage = HTML5 code + CSS code**



# BENEFITS OF USING CSS



- Code will be reused, as the same style may apply to many elements in many pages
- You can create multiple style sheets and hence have different skins for your website that can change how the website looks and even behaves.
- Editing styles and/or finding and fixing styling errors will be easier, as styles are centralized .
- Content and display are separated so the data becomes easier to read automatically.

**Webpage = HTML5 code + CSS code**



# TOPICS

- What is CSS?
- **Levels of CSS**
- Selectors
- Property Values
  - Fonts/Colors
  - Links
  - Lists
  - Alignment of text
  - Box Model
  - `<span>` vs `<div>`
- Bootstrap

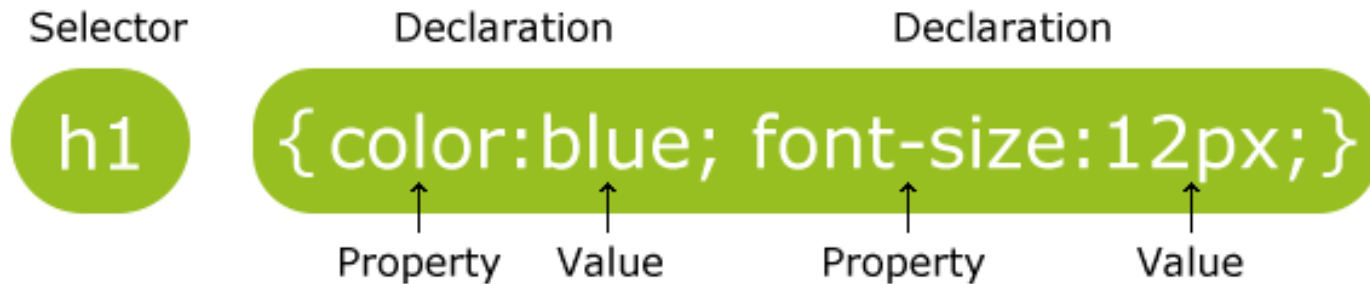


# LEVELS OF STYLE SHEETS

- **Inline**
  - Given for a specific occurrence of a tag and apply only to that tag
  - defeats the purpose of style sheets - uniform style
  - W3C deprecated inline style in 2001
- **Document-level style sheets**
  - apply to the whole document in which they appear
  - Styles defined in the head section
- **External style sheets**
  - can be applied to any number of documents
  - Specify style sheet to use in the head section
- An external style sheet can be written in any text editor. The file must not contain any HTML code, and must be saved with a .css extension.



# CSS SYNTAX



- Selector = HTML element want to style
- Declaration
  - property and value
  - Always ends with a semicolon (;)
  - Example: [CSS Ex1.html](#)
- CSS Comments in the form /\* ..... \*/





# CSS SYNTAX EXAMPLE

CSS\_Ex1.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title> CSS Example 1 </title>
    <style>
      /* This is CSS*/
      body {background-color:pink;}
      h1 {font-size:30pt;color:yellow;text-align:center;}
      h2 {color:purple;}
      p {margin-left:10px;}
    </style>
  </head>
  <body>
    <h1>This header is 30 pt</h1>
    <h2>This header is purple</h2>
    <p>This paragraph has a left margin of 100 pixels</p>
  </body>
</html>
```



# EXAMPLES

## 1. Inline

Use this method sparingly

```
<p style="color:red;margin-left:20px;">  
  This is a paragraph.  
</p>
```

## 2. Document level

Selector

```
<head>  
  <style type="text/css">  
    h1 {color:red;}  
    p {margin-left:20px;}  
    body {background-image:url("images/back40.gif") ;}  
  </style>  
</head>
```



# EXAMPLE (CONTINUED)

## 3. Using an external Style Sheet

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

Relationship: link to an  
external sheet

- Inside the style sheet: mystyle.css

```
h1 {color:red;}  
p {margin-left:20px;}  
body {background-image:url("images/back40.gif");}
```



# TOPICS

- What is CSS?
- Levels of CSS
- **Selectors**
- Property Values
  - Fonts/Colors
  - Links
  - Lists
  - Alignment of text
  - Box Model
  - `<span>` vs `<div>`
- Bootstrap



# SIMPLE/ELEMENT SELECTOR

- tag names {property\_1:value\_1;property\_2:value\_2;...}

ex: `h1 {color: white;}`

selector	h1
declaration	{color:white;}
property	color
value	white

- Selectors can be grouped as in

`h1, h2, h3 {color: green;}`

- Declarations can be grouped as in

`h1 {color: white; background-color: black;}`



# SIMPLE/ELEMENT SELECTOR

CSS\_Ex2.html

```
<!DOCTYPE html>
<html lang = "en" >
<head>
    <style>
        p.center {
            text-align: center;
            color: red;
        }
    </style>
</head>
<body>
    <h1 class="center">This heading will not be affected</h1>
    <p class="center">This paragraph will be red and center-aligned.</p>
    <p>A paragraph with no style</p>
</body>
</html>
```



# CLASS SELECTORS

```
p.normal {property-value list}  
p.warning {property-value list}
```

```
<p class = "normal">  
  A paragraph in 'normal' presentation style  
</p>
```

```
<p class = "warning">  
  A paragraph in 'warning' presentation style  
</p>
```

```
<p>  
  A paragraph with no style  
</p>
```



# GENERIC SELECTORS

```
.sale {property-value list}
```

```
<h1 class = "sale"> Weekend Sale </h1>  
<p class = "sale"> ... </p>
```

[http://www.w3schools.com/Css/tryit.asp?filename=trycss\\_syntax\\_class](http://www.w3schools.com/Css/tryit.asp?filename=trycss_syntax_class)





# ID SELECTORS

```
#section14 {font-size: 20;} ✓  
h2#section14 {font-size: 20;} ✓  
p#section14 {font-size: 20;} ✗
```

```
<h2 id = "section14"> 1.4 Calico Cats </h2>
```

```
#section14 {font-size: 25; color: red;} ✗  
h2#section14 {font-size: 25; color: blue;} ✓
```

```
<h2 id = "section14"> 1.4 Calico Cats </h2>
```



# SELECTORS: ON THEIR OWN, GENERIC & ID

```
<head>
  <meta charset="utf-8"><title> CSS Example 3 </title>
  <style>
    body {background-color:pink;}
    h1 {font-size:XX-large;color:yellow;text-align:center;}
    h2 {font-size:x-large;color:purple;}
    p {margin-left:10px;}
    .warning {color:red;}
    #section14 {color:white;}
  </style>
</head>
<body>
  <h1>This header is xx-large pt</h1>
  <h2>This header is x-large & purple</h2>
  <h2 id = "section14"> section 14 with id</h2>
  <h2 class = "warning"> section 14 with warning</h2>
  <p >This paragraph has a left margin of 100 pixels</p>
</body>
```

[CSS Ex3.html](#)



# TOPICS

- What is CSS?
- Levels of CSS
- Selectors
- Property Values
  - **Fonts/Colors**
  - Links
  - Lists
  - Alignment of text
  - Box Model
  - `<span>` vs `<div>`
- Bootstrap



# HOW TO GIVE PROPERTY VALUES

- *Length* - numbers, maybe with decimal points

- Units:

- Not Scalable:

- px – pixels
    - pt – points

	body { font-size: 100%; }	body { font-size: 120%; }
font-size: 1em	The quick brown fox	The quick brown
font-size: 12pt	The quick brown fox	The quick brown fox
font-size: 16px	The quick brown fox	The quick brown fox
font-size: 100%	The quick brown fox	The quick brown

© KyleSchaeffer.com

- Scalable

- em – 1em is equal to the current font size.
    - % - % of the current font size

**Note:** No space between the number & the unit

1.5em ✓

1.5 em ✗



# HOW TO GIVE PROPERTY VALUES ...

- Colors:

1. Color name: red green pink ...
2. `rgb(n1, n2, n3)`  
Numbers can be decimal or percentages
3. Hexadecimal form: `#XXXXXX`

- Handy Sites for colors:

- [http://www.w3schools.com/cssref/css\\_colors.asp](http://www.w3schools.com/cssref/css_colors.asp)
- <http://paletton.com/#uid=14B110klHm3kb8Clzg2lLrYmDCO>



# EXAMPLE

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      #p1 {background-color:#ff0000;} //Red
      #p2 {background-color:#00ff00;} //Green
      #p3 {background-color:#0000ff;} //Blue
      #p4 {background-color:#ffff00;} //Yellow
      #p5 {background-color:rgb(255,0,255);}</style>
    </head>
    <body>
      <p>HEX colors:</p>
      <p id="p1">Red</p>
      <p id="p2">Green</p>
      <p id="p3">Blue</p>
      <p id="p4">Yellow</p>
      <p id="p5">Cerise</p>
    </body>
  </html>
```



# FONT PROPERTY: FONT-FAMILY

```
font-family: Arial, "Arial Black", Futura, serif
```

- Value is a list of font names - browser uses the first in the list it has
- **Generic fonts:** `serif`,  
`sans-serif`  
**and** `monospace`  
(defined in CSS)

`font-family:serif`

`font-family:sans-serif`

*font-family:cursive*

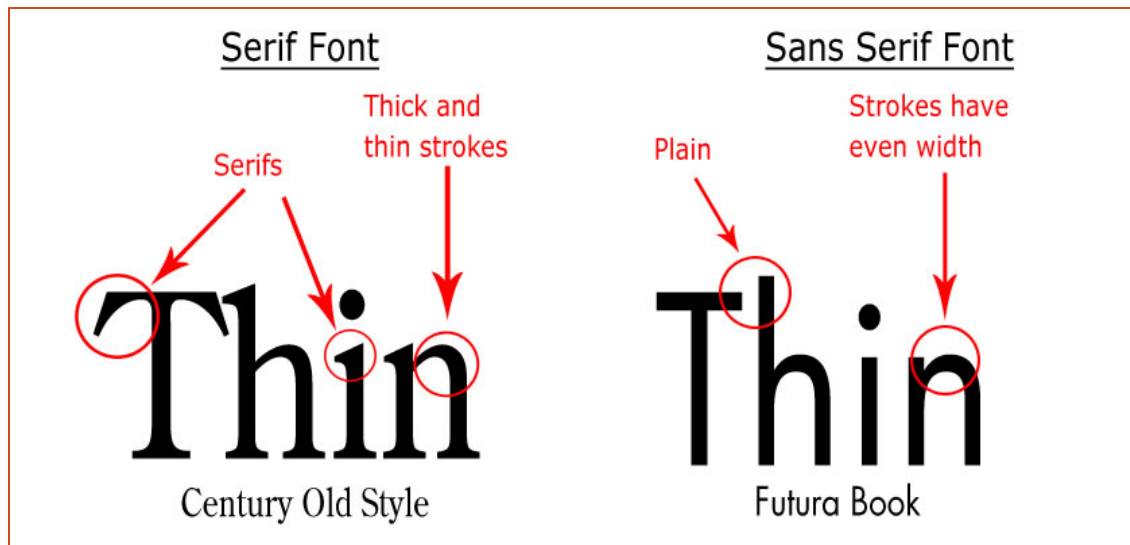
`font-family:fantasy`

`font-family:monospace`



# FONT-FAMILY: EXAMPLES

- Samples of serif typefaces
- Samples of sans-serif typefaces
- Samples of cursive typefaces (let you play ...)
- Samples of monospaced typefaces





# FONT PROPERTY: FONT-SIZE

```
font-size: 10pt
```

- Possible values: a length number (px, pt), a %, an em or a name, such as xx-small, x-small, small, medium, large, x-large **and** xx-large, etc

## Font Property: font-variant

```
font-variant: small-caps
```

- Default is normal, but can be set to small-caps



## Font Property: font-style

```
font-style: italic
```

- Possible values: italic, oblique (useless), normal

## Font Property: font-weight

```
font-weight: bolder
```

- Possible values: bolder, lighter, bold, normal
- Could specify as a multiple of 100 (100 – 900) (400 is normal, 700 is bold)

## Font Shorthand: font

```
font: italic bolder 14pt Arial, serif
```

- Order must be: style, weight, size, name(s)

## Font Property: text-decoration

```
text-decoration: line-through
```

- Possible values: line-through, overline, underline, none

# TOPICS

- What is CSS?
- Levels of CSS
- Selectors
- Property Values
  - Fonts/Colors
  - **Links**
  - Lists
  - Alignment of text
  - Box Model
  - `<span>` vs `<div>`
- Bootstrap



# LINK STYLES

- Can style links differently depending of state in (color, background etc...)
- Link states:
  - `a:link` - a normal, unvisited link
  - `a:visited` - a link the user has visited
  - `a:hover` - a link when the user hovers over it
  - `a:active` - a link the moment it is clicked



# LINK STYLE: EXAMPLE 1

```
<head>
```

```
  <style>
```

```
    a:link {color:#FF0000;}      /* unvisited link */
```

```
    a:visited {color:#00FF00;}  /* visited link */
```

```
    a:hover {color:#FF00FF;}    /* mouse over link */
```

```
    a:active {color:#0000FF;}   /* selected link */
```

```
  </style>
```

```
</head>
```

LinkDemo.html

## Note:

1. a: hover **MUST** come after a:link and a:visited
2. a:active **MUST** come after a:hover



# LINK STYLE: EXAMPLE 2

- To remove the underline:

```
a:link {text-decoration:none;}
```

- A more elaborate example:

[http://www.w3schools.com/css/tryit.asp?filename=trycss\\_link2](http://www.w3schools.com/css/tryit.asp?filename=trycss_link2)



# WHAT IS COLOR & STYLE OF HEAD HEADER?



## colors.css

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

## color.html

```
<link rel="stylesheet" type="text/css" href="colors.css" />
  <style type="text/css" >
    h1.pink {color:hotpink;}
    h1.normal{font-style:normal}
  </style>
<body>
  <h1> Heading 1 </h1>
  <h1 class="pink"> Heading 2</h1>
  <h1 class="normal"> Heading 3</h1>
</body>
```



# WHAT IS COLOR & STYLE OF HEAD HEADER?

## colors.css

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



## color.html

```
<link rel="stylesheet" type="text/css" href="colors.css" />
<style type="text/css" >
  h1 {color:limegreen;}
  h1.pink {color:hotpink;}
  h1.normal{font-style:normal}
</style>
<body>
  <h1> Heading 1 </h1>
  <h1 class="pink"> Heading 2</h1>
  <h1 class="normal"> Heading 3</h1>

</body>
```

# TOPICS

- What is CSS?
- Levels of CSS
- Selectors
- Property Values
  - Fonts/Colors
  - Links
  - **Lists**
  - Alignment of text
  - Box Model
  - `<span>` vs `<div>`
- Bootstrap



# List Properties: list-style-type

- Possible values for unordered lists: disc, square, circle or none

```
<style type = "text/css">  
    ul {list-style-type: square;}  
</style>
```

[http://www.w3schools.com/cssref/playit.asp?filename=playcss\\_ul\\_list-style-type&preval=disc](http://www.w3schools.com/cssref/playit.asp?filename=playcss_ul_list-style-type&preval=disc)

[https://www.w3schools.com/css/tryit.asp?filename=trycss\\_list-style-type\\_ex](https://www.w3schools.com/css/tryit.asp?filename=trycss_list-style-type_ex)

# List Properties: list-style-type

- Possible values for ordered lists: decimal, decimal-leading-zero, lower-alpha, upper-alpha, ...

```
<style type = "text/css">  
    ol {list-style-type: upper-roman;}  
</style>
```

[http://www.w3schools.com/cssref/playit.asp?filename=playcss\\_ol\\_list-style-type&preval=armenian](http://www.w3schools.com/cssref/playit.asp?filename=playcss_ol_list-style-type&preval=armenian)

# WHAT DO YOU THINK...?



```
<style type = "text/css">
    ol {list-style-type: upper-roman;}
    ____ {list-style-type: upper-alpha;}
    ____ {list-style-type: decimal;}
</style>
```

- Fill the **first** blank line with a proper selector

- A. ol
- B. li
- C. ol ol
- D. ol, li
- E. ol ol ol

- I. First level list - item 1
  - A. Second level list - item 1
    - 1. 3rd level list - item 1
    - 2. 3rd level list - item 2
  - B. Second level list - item 2
    - 1. 3rd level list - item 1
    - 2. 3rd level list - item 2
- II. First level list - item 2
  - A. Second level list - item 1
    - 1. 3rd level list - item 1
    - 2. 3rd level list - item 2
  - B. Second level list - item 2
    - 1. 3rd level list - item 1
    - 2. 3rd level list - item 2



# WHAT DO YOU THINK...?



```
<style type = "text/css">
    ol {list-style-type: upper-roman;}
    ____ {list-style-type: upper-alpha;}
    ____ {list-style-type: decimal;}
</style>
```

- Fill the **second** blank line with a proper selector

- A. ol
- B. ol ol
- C. ol, li
- D. li li
- E. ol ol ol

- I. First level list - item 1
  - A. Second level list - item 1
    - 1. 3rd level list - item 1
    - 2. 3rd level list - item 2
  - B. Second level list - item 2
    - 1. 3rd level list - item 1
    - 2. 3rd level list - item 2
- II. First level list - item 2
  - A. Second level list - item 1
    - 1. 3rd level list - item 1
    - 2. 3rd level list - item 2
  - B. Second level list - item 2
    - 1. 3rd level list - item 1
    - 2. 3rd level list - item 2