

CMSC335

Web Application Development with JavaScript



CSS

Department of Computer Science
University of MD, College Park

Slides material developed by Ilchul Yoon, Nelson Padua-Perez

Miscellaneous

- **VS Code Settings**
 - Select the wheel (bottom left) and then "Settings"
 - Auto Save with **afterDelay** option automatically saves every second
 - » Recommended
- **Hard Refresh/Hard Reload** - clears browser cache for a specific page, forcing a loading of the most recent version of the page. Note: "Refresh" arrow on your browser address bar is not a hard refresh
 - Windows - hold down Ctrl, Shift, and press R
 - Mac - hold down Cmd, Shift, and press R
- **Soft Refresh** - reloads the page from the cache
- **Commenting out code in VS Code**
 - Ctrl and forward slash turns a line into a comment (or vice versa). Works for HTML, CSS, JavaScript
- Above information also available at
- <https://www.cs.umd.edu/~nelson/classes/resources/web/> (under IDE/Editors)

CSS (Cascading Style Sheets)

- Official W3C standard **for controlling presentation**
- Style Sheets
 - Text file with rules
 - They include no HTML
 - Use a **.css** extension
 - Allow you to apply typographic styles (font size, line spacing, etc.)
 - Allow you to have page layout control
 - Allow smaller HTML files by avoiding redundancy in style specification
 - Allow easy update of a collection of pages by updating only a single file
- Why CSS?
 - <http://www.csszengarden.com/>

CSS

- Rule
 - Basic element of a style sheet
 - **Describes the formatting associated with a page element**
- Rule format

selector { declaration block }
- **Selector:** identifies what should be styled in a web document
 - e.g., h1, p
- **Declaration**
 - Describes styling information (what and how that portion of the web document should be modified)
 - **Example:** color: blue
 - **Declaration format:** property (e.g., color), colon (:) and value (e.g., blue)
- **Declaration block:** declarations enclosed in { } and separated by semicolons (last one does not need a semicolon, but it is OK to have it)
- **Example:** h1 {color: blue; text-align: center;}
- **Properties list:** <http://www.w3schools.com/cssref/>

Types of Style Sheets

- **Inline**

- Style information applied to a specific tag
(e.g., `<p style="font-size: 2em;">New Semester</p>`)
- Avoid if possible

- **Internal**

- Using the `<style>` tag in the header of the HTML document
- Convenient to provide own style to a specific page
 - » It can override an include style sheet
- **Example:** InternalStyle.html

- **External**

- External style sheet which web pages link to (see `<link>` tag)
- Preferred approach

- **Example:** ExternalFile.html, ExternalFile.css

- Files do not need to have the same base name

- We can use `/*` multi-line `*/` to comment CSS

CSS Validator

- <http://jigsaw.w3.org/css-validator/>
- Notice you have three choices
 - By URI
 - By file upload
 - By direct input
- Look at the “More Options” section
- We can also validate using the VS Extension you installed for HTML
 - Let’s see a demo

Regarding Size

- Two types of length units:
 - **Relative** (e.g., em, rem, %)
 - » **Example:** 3em
 - » **Example:** 4rem
 - **Absolute** (e.g., px, cm, pc)
- For the course: we will mainly use pixels, em, rem, %
- Relative
 - **em:** Relative to the size of the current font
 - » 3em: 3 times the size of the current font
 - **rem ("root em"):** inherits its value from the base font size of the **root element**
 - » In HTML, the root element is the **<html>** element
 - » **rem** is constant throughout a document
 - The **em** and **rem** units are practical in creating a scalable layout
 - <http://www.sitepoint.com/power-em-units-css/>
- **Whitespace cannot appear between the number and the unit**
 - **correct:** 3em, **incorrect:** 3 em
 - If the value is zero, the unit can be omitted

Regarding Colors

- You can specify colors using predefined colors (e.g., **yellow, white, red, purple, blue**)
- You can specify a color by indicating the red, green, and blue components. For example, all the following are equivalent:
 - red
 - `rgb(255,0,0)`
 - `#ff0000` or `#f00` (shorter, cannot use it for all colors)
- RGBA adds an alpha channel that allows us to specify the opacity for a color
 - RGBA color specification: **`rgba(RED, GREEN, BLUE, ALPHA)`**
 - Alpha value of 0 (transparent), alpha value of 1.0 (opaque)
 - » **Example:** `rgba(125, 125, 125, 0.5);`
- In VS Code, you can define a particular color (including opacity). Hover over a color of a CSS declaration in VS Code to see options
- References:
 - http://www.w3schools.com/css/css3_colors.asp

Kinds of Selectors

- **Type Selectors:**
 - Based on the name of an HTML tag (e.g., p, table, etc.)
 - `p { color: red; }`
- **Class Selectors:**
 - Allow us to define a style (“type of style”)
 - » Allow us to apply the same rules to a set of elements
 - Created with a period . (also known as a full stop)
 - Applied to an element using the **class** keyword
 - You can apply several class selectors to an element
 - **Example:** ClassIdSelectors.html
- **ID Selectors:**
 - Similar to class selectors but appear only once in the document
 - Used when you need to apply a style only once in your document, or you want to capitalize on an id that already exists
 - Created using #
- **Example:** ClassIdSelectors.html

Kinds of Selectors

- **Pseudo-class** - "phantom class" that allows assigning style to elements based on the **state of an element** (e.g., a link has been clicked) or markup patterns (e.g., odd rows of a table)
 - Attached to selectors (e.g., p:**hover**)
 - Four popular pseudo-classes uses are
 - » **a:link**: the initial color of a link
 - » **a:visited**: color for a visited link
 - » **a:hover**: color when the mouse hovers over the link
 - » **a:active**: color during the clicking of the link
- **Example:** Pseudoclass.html

About Properties

- Inheritance of Properties:
 - A child element inherits the same properties of its parent element (unless otherwise indicated by the CSS specification)
 - **Examples of inherited properties**
 - » color, font-family, font-size, letter-spacing, ...
 - **Examples of non-inherited properties**
 - » border, margin, padding, ...
- Properties can overwrite others
 - **Example:** Overriding.html
- “Cascading” in Cascading Style Sheets
 - Determines the final style for an element when multiple can apply. The “cascade” refers to combining different style sources

font-family

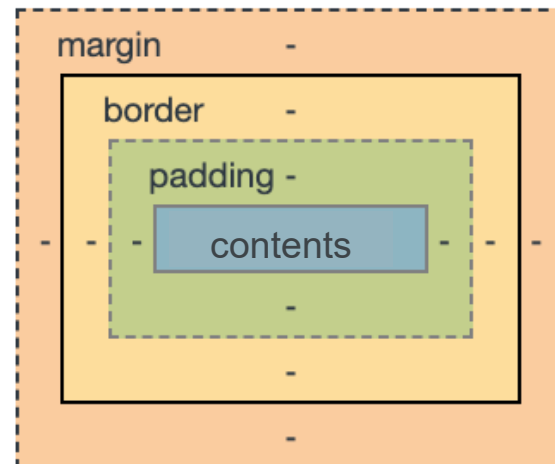
- In a document we usually associate with the <body> tag a font-family
- The font-family property defines the fonts to use (in order)
- Two generic font families used as default values: serif and sans-serif
 - **serif**
 - » Line or stroke added to letter or symbol
 - » **Examples:** Times New Roman, Georgia
 - **sans-serif**
 - » No serif (no line or stroke added to letter or symbol)
 - » **Examples:** Verdana, Arial
- Additional generic font families
 - **monospace**
 - » **Example:** Courier New, Consolas
 - **cursive** - *Lucida Handwriting*
 - **fantasy** - Comic Sans MS, ...
- **Example:** FontFamily.html

Lorem Ipsum

- Lorem ipsum - placeholder text that serves to fill a space without actually saying anything
- Main reason for its use is, so people don't focus on the text
- It is rooted in Latin
- Comes from sections 1.10.32 and 1.10.33 of "de Finibus Bonorum et Malorum" (The Extremes of Good and Evil) by Cicero
- Reference: <https://www.lipsum.com/>
 - You can generate lorem ipsum lists, paragraphs, etc., using the previous site
- Use `<html lang="zxx">` (HTML validation will fail otherwise)

Box Model

- CSS box model:
 - Box around an HTML element with the following components: margin, border, padding, content
- Allow us to define space between different elements
- Four sections can be identified:
 - **contents**: what lies in the middle of the box (text, image, etc.)
 - **padding**: surrounds the content
 - **border**: surrounds the padding and represents the box border
 - **margin**: surrounds the border



Box Model (Padding/Margin)

- You can adjust individual padding/margin properties by using:
 - padding-bottom, padding-left, padding-top, padding-right
 - margin-bottom, margin-left, margin-top, margin-right
- The margins, borders, paddings, and background properties of block elements (e.g., body, p, etc.) are not passed to the children
 - i.e., not inherited
- Box width
 - left + right padding, left + right border, left + right margin, content width
 - **width** property - sets the **content** width
- Box height is determined in the same way
 - **height** property - sets the **content** height
- **Example:** BoxModel.html

Shorthand Property

- Allows you to specify several properties by using only one
 - If you don't specify one of the properties, a default value will be used
- Commonly used shorthand properties
 - background
 - font
 - list-style
 - margin
 - border
 - padding
- **Example:** ShorthandProperties.html

Background properties

- Background properties can be applied to <p>, <div>, <body>, and other elements
- **background-color** - color
- **background-image** - location (URL) of image
- **background-repeat** - how the image repeats
 - Possible values for repetition
 - » **no-repeat** - one instance of the image
 - » **repeat** - tile
 - » **repeat-y** - repeats on the y-axis
 - » **repeat-x** - repeats on the x-axis
- **background-attachment**
 - Indicates attachment of the image to the containing element
 - Possible values are:
 - » **scroll**: default value
 - » **fixed**: the image will stay stationary as the scrolling takes place

Background (cont.)

- **background-position**
 - Possible values (some combinations are valid)
 - » top, bottom, center, left, right
- **background-size**
 - Possible values
 - » auto auto - retains the original size
 - » 000px 000px
 - » x% y%
 - » contain | cover | ...
- There is a shorthand property for backgrounds
 - **background: lightblue url("campusBldg.jpg") no-repeat fixed center;**
- **Examples:** Background.html

Formatting Lists and Tables

- **Example:** ListsFormatting.html
- **Examples:** TablesFormatting1.html, TablesFormatting2.html

Size: rem vs em

- **Examples:** SizeRemVsEm.html