

Announcements

- ❖ No posting of code in the forum
- ❖ Check class announcements daily

Types of Style Sheets

❖ **Inline**

- ❖ Style information applied to specific tag (e.g., `<p style=..."`).
- ❖ Avoid if possible.

❖ **Internal**

- ❖ Using the `<style>` tag in the header of the html document.
- ❖ Convenient to provide own style to a specific page.
- ❖ Example: `InternalStyle.html`

❖ **External**

- ❖ External style sheet which web pages link to (see `<link>` tag).
- ❖ Preferred approach.
- ❖ Example: `ExternalFile.html` and `ExternalFile.css`

CSS

- ❖ Why cascading?
 - ❖ Rules can come from different sources (inline, external file, etc.). The final set of rules that apply to a document comes from cascading all the sources
- ❖ Rule Conflict Resolution
 - ❖ To resolve conflicts, styles defined at a specific level override those set at a higher level
 - ❖ Example: you can set the color of body text to be blue but you can override to red the text in a list
 - ❖ When multiple style files are linked or imported the last will take precedence
- ❖ A child element inherits the same properties of its parent element (unless otherwise specified)

CSS Validator

- ❖ <http://jigsaw.w3.org/css-validator/>
- ❖ Notice you have three choices
 - ❖ by URI
 - ❖ by File Upload
 - ❖ by direct input

Colors

- ❖ You can specify colors using one of the following predefined colors:
yellow, white, teal, silver, red, purple, orange, olive, navy, maroon, lime, green, gray, fuchsia, blue, black, aqua
- ❖ Source for colors
http://www.w3schools.com/html/html_colors.asp
- ❖ 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

CSS Rules Review

- ❖ CSS rules has two parts:
 - ❖ selector – what part of the web page gets styled
 - ❖ property:value pairs – what to change and the nature of the change

Example:

```
h2 { font-size: 120%; color: blue; }
```

selector

*Property
:value*

*Property
:value*

About CSS

- ❖ **Comments** – Surrounded by `/* */`
- ❖ **Inheritance of Properties** - A child element inherits the same properties of its parent element (unless otherwise indicated by the CSS specification)

Kinds of Selectors

- ❖ **Type Selectors** – Those based on the name of an HTML tag (e.g., p, table, etc.)
 - ❖ p { color: red; }
- ❖ **Pseudo-classes** – attached to selectors to specify a state. Four popular pseudo-classes are:
 - ❖ a:link – initial color of a link
 - ❖ a:visited – color for a visited link
 - ❖ a:hover – color when mouse hover over link
 - ❖ a:active – color during the clicking of the link
- ❖ **Example:** Selectors.html, Selectors.css

Kinds of Selectors

- ❖ **Class Selectors** – Allow us to apply the same CSS rules to different elements
 - ❖ Use when you need to apply a style many times in your document.
 - ❖ Created with a period (also known as full stop)
 - ❖ **Example:** Selectors.html, Selectors.css
- ❖ **ID Selectors** – Like class selectors but appear only once in the document
 - ❖ Used when you need to apply a style only once in your document.
 - ❖ Created using #
 - ❖ **Example:** Selectors.html, Selectors.css
- ❖ **Others** (will see them later on)
 - ❖ Descendant Selectors, child selectors, attribute selectors, universal selectors

Additional HTML Elements

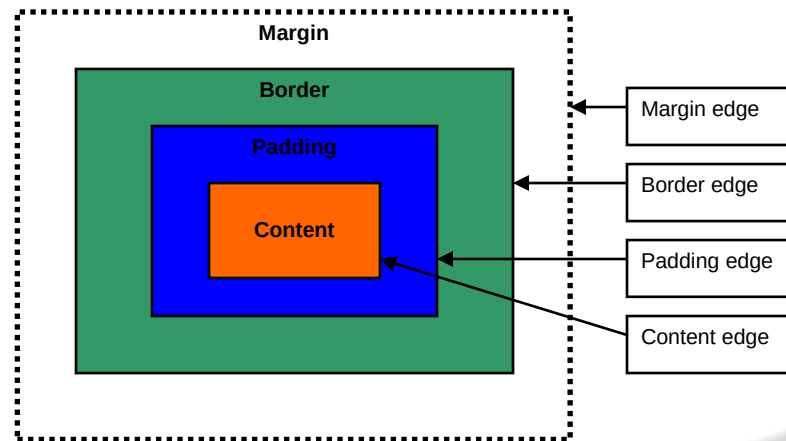
- ❖ `<div>` and `<span>` -
 - ❖ Allow you to delimit a section of the HTML body
- ❖ `<span>`
 - ❖ Used to wrap inline content (e.g., text sequence)
- ❖ You can apply style to the sections defined by `span` and `div`
- ❖ The following example divide the document in three main areas
- ❖ **Example:** `SpanDiv.html`, `SpanDiv.css`

Submit Server

- ❖ What is the submit server?
- ❖ How are you going to use it?
- ❖ How to create the zip (compress) file?
 - ❖ In Mac → Control→Click→Compress “FileName”
 - ❖ In Windows → Right Click→ Send To → Compressed (zipped) folder
- ❖ You can submit as many times as you want (last submission graded)
- ❖ Use the submit server to store backup copies and to show code during office hours
- ❖ Do not wait until the day the project is due to access the submit server (no extensions will be granted due to submit server access problems)

Box Model

- ❖ Each block element (e.g., p) contains four edges (top, bottom, right, and left) defining a box
- ❖ Four sections can be identified with a block element
 - ❖ Content – 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



<div> is a block element

- ❖ <div> Defines a block-level entity
 - ❖ Browser starts a div element's content on its own line.
- ❖ <body> also defines a block-level entity
- ❖ You can use your box model knowledge to add more style to your pages
- ❖ **Example:** BoxModel1.html, BoxModel1.css

Box Model

- ❖ Let's explore more of the box model with the following examples
- ❖ **Example:** BoxModel2.html, BoxModel2.css
- ❖ **Example:** Padding.html, Padding.css
- ❖ The margins, borders, padding, and background properties of block elements (e.g., body, p, etc.) are not passed to their child block-level elements

Setting Size

- ❖ Percentages – size of the font is based on the size of the parent element
- ❖ Length units
 - ❖ centimeters (cm)
 - ❖ millimeters (mm)
 - ❖ points (pt) - 1 pt → 1/72 inch
 - ❖ picas(pc) - 1pica → 12 pts
 - ❖ inches (in)
- ❖ Relative
 - ❖ ex – height of the lowercase x in the font
 - ❖ px – pixels
 - ❖ em –
 - ❖ Corresponds to the value of the font-size property of the element on which is used. If it appears in the font-size property itself it refers to the font-size of the parent element.
 - ❖ If the parent is body tag and no font size is specified then the size is looked in the user's preferences specified in the browser
 - ❖ Allows you to define scalable style sheets
 - ❖ Preferred unit

Font-size Keywords

- ❖ xx-small
- ❖ x-small
- ❖ small
- ❖ medium
- ❖ large
- ❖ x-large
- ❖ xx-large
- ❖ larger
- ❖ smaller
- ❖ **Example:**
FontsizeKeywords.html, FontsizeKeywords.css