CIS 155 Notes

* Tree Concept (HTML)
  + Every HTML file can be viewed as an (upside down) tree
    - Have a “Main” element with…
      * Nested Elements that are Children of the “Main” Element
    - Can think of it as a family tree…
      * A diagram of a diagram

        Description automatically generated with medium confidenceGrandparent, parent, children, grandchildren
* Relational Selectors (CSS)
  + How to code a Child Element
    - Parent-Element > Child-Element { }

A black text on a white background

Description automatically generated

* + How to code a General Sibling
    - Tags within the same “indentation”
    - Sibling-Tag-Above ~ Sibling-Tag-Below { }
      * Only affects the Sibling tags below it

A close up of a sign

Description automatically generated

* + How to code a Descendant
    - Affects tags anywhere down the tree from the “Main” tag
    - Main-Tag Descendant-tag { }

A close up of a text

Description automatically generated

* + How to code Adjacent Sibling
    - Affects the tag that comes right after the “Main” tag
    - Main-Tag+Next-Tag { }

A close up of a sign

Description automatically generated

A white screen with black text

Description automatically generatedUse this to as a reference to understand the stuff above

* How to code Combination Selectors
  + A Selector for a class within an element
    - Combines Tag Names with Classes
    - Tag.Class { }

A black text on a white background

Description automatically generated

* + Multiple Selectors
    - Put commas between the tags you want to select
    - Tag, tag, tag { }
    - Second example in picture reads as
      * All p tags and li descendants of ul.speakers

A text on a white background

Description automatically generated

Use this to help understand the examples above

A computer screen shot of a program

Description automatically generated

* How to code Attribute Selectors
  + Attributes are anything that come right after the tag name followed by an “=”
  + Attribute is specified in the code in [x]

A white background with black text

Description automatically generated

Use this to help understand the above

A white screen with text

Description automatically generated

* Common Pseudo-Classes & Pseudo-Elements
  + Pseudo-Classes
    - A close-up of a text

      Description automatically generatedNot necessarily coded in the HTML but they kind of represent the state of your webpage
  + Pseudo-Elements
    - A close up of a text

      Description automatically generatedWorks like Pseudo-Classes
* How do Cascade Rules Work?
  + What happens when styles applied to an element conflict?
    - 1: Style rule w/ highest specificity selector wins!
    - 2: If same specificity, style rule applied last wins
  + If multiple style sheets, the last one will override previous
  + User style sheets: Users can create special style sheets for their browsers
    - People w/ poor vision could enlarge the font sizes
  + If you want to improve the precedence of a style rule put a “!important” at the end

A close up of a sign

Description automatically generated

* How to create & use custom properties (variables)
  + CSS Variables

A black background with white text

Description automatically generated

A screenshot of a computer code

Description automatically generated

* + When declaring a custom property
    - Put it at the top of the document
    - Syntax
      * { --variable: value; }
  + To use a custom property
    - Syntax EX:
      * { color: var(--custom\_property); }

Need this for Module 4 CYOW Assignment

Github Repository Link

<https://github.com/I-AJ-T/my-website>

Github Pages Link

<https://i-aj-t.github.io/my-website/>

* Type Selector
  + Lines:
    - 11 (header)
    - 17 (p)
* ID Selector
  + Lines:
    - 4 (#logic\_pic)
    - 32 (#summon\_text)
* Class Selector
  + Lines:
    - 27 (.heading\_text)
    - 56 (.section\_heading\_text)
    - 74 (.important\_text)
    - 80 (.important\_text\_1)
    - 86 (.important\_text\_2)
* Relational Selector
  + Lines
    - 23 (header > h1)
    - 38 (p#summon\_text a)
* Combination Selector
  + Lines
    - 50 (h1.overview\_text)
    - 64 (p.list\_heading)
    - 68 (ul, ol li)
* Pseudo-Class
  + Line
    - 43 (p#summon\_text a:hover)
* Pseudo-Element
  + Line
    - 60 (.section\_heading\_text::first-letter)

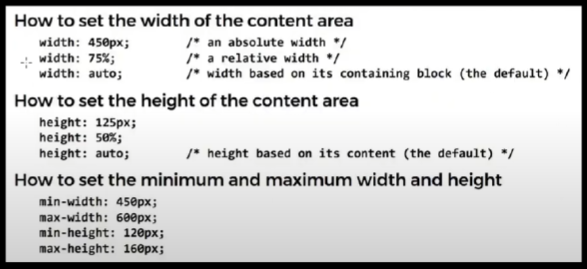
Mod 5 Lecture Notes:

* CSS Box Model
  + Web browsers place each block element in a box
  + Margin
    - Space between border and edge of box
  + Padding
    - Space between content area and border

A diagram of a rectangular object with arrows

Description automatically generated

* How to set Widths & Heights
  + Auto is the default for width & height if you don’t set it
  + Can use relative or absolute measurements for width & height
  + Min- & max- width & height will become useful later when we learn more about Responsive Web Design



* How to set Margins
  + Can set margins individually or all at once
  + For setting margin property (which sets all 4 directions)
    - If you specify 1 value
      * It’s all 4 sides
    - If you specify 2 values
      * It’s top/bottom & then left/right
    - If you specify 3 values
      * It’s top, left/right, bottom
    - If you specify 4 values
      * It’s clockwise from the top
  + Margins for 2 elements on top of each other collapse
    - Meaning only larger margin counts

A screenshot of a computer

Description automatically generated

* How to set Padding
  + Same as Margins, except padding never collapses

A screenshot of a computer

Description automatically generated

* Using a “Reset Selector”
  + A universal selector that sets all margins & padding to 0
  + Useful for getting rid of all default margins/padding that elements might have so you can code your own

A yellow rectangular sign with black text

Description automatically generated

* How to set borders
  + Lots of different ways to customize borders
  + You can add borders to a single side of a box, just some sides, or to all sides

A white text on a white background

Description automatically generated

A white background with black text

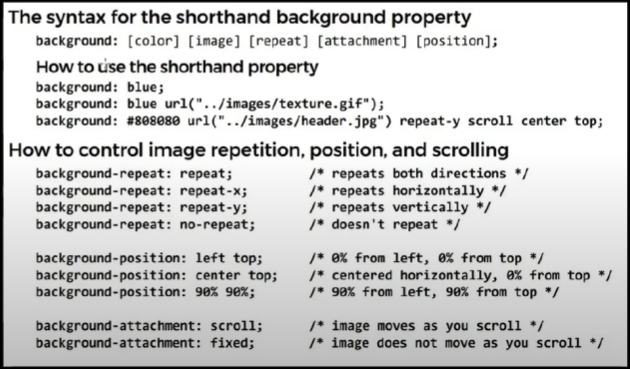
Description automatically generated

* How to set rounded corners & shadows to borders
  + Border-radius
    - Used for rounding corners
    - Radius of a circle defines the rounded edge
  + Box-shadow
    - Is similar to text-shadow
  + This takes some experimenting to find the right parameters

A screenshot of a computer program

Description automatically generated

* How to set background colors & images
  + Can set a background color or image for any box
  + Default for background image is that it will repeat horizontally & vertically & tile across the box
  + Web Accessibility Note
    - You have to make sure a background image doesn’t make text harder to read



* How to set background gradients

A screenshot of a computer program

Description automatically generated

This is for Mod 5 CYOW Assignment

* Added Background Image to body element on HTML (index.html) line 13 and CSS line 6
* Added Margin, Border, and Gradient to header element on HTML (index.html) line 14 and CSS line 60
* Added Border to img element on HTML (index.html) line 15 and CSS line 47
  + This had Margins on it before this assignment
    - The same goes for the marquee “IMPORTANT” text HTML (index.html) line 110 and CSS line 138
* Added Margin, Border, Padding, and Gradient to main element on HTML (index.html) line 28 and CSS line 18
* Added Margins to footer element on HTML (index.html) line 192 and CSS line 39
* The second page link is on HTML (index.html) line 54
  + Article element for columns is on HTML (pvp\_pve\_page.html) line 30 and CSS line 33
  + The second page uses the same style sheet as the index.html page and has the same margins, borders, gradients, and background image added to it
  + The second page is also set up for more information to be added to it and a link to go back to the home page
    - I am just mocking up what I think I want the other pages to potentially look like for when we start adding those and the navigation bar. Hopefully I am on the right track and doing this in a “correct” way

Navigation Menu stuff is in file Module\_5\_Example\_Video\_2

Notes for peer review assignment

* Alignment
  + Good alignment
    - Elements line up nicely together
* Proximity
  + Things that are closely related are in close proximity of each other
* Repetition
  + Pages have relatively the same layout or styling
* Contrast
  + Text is readable, elements are easily visible
* Questions to ask:
  + Is text concise & easy for a reader to quickly find on the page?
  + Is critical info “above the fold”?
  + Does the homepage “sell” the website well?

For my website fix the content to fit before the fold when pages are loaded. Especially for the home page.

Fix images and videos to fit w/in a viewport of a browser