CSS:

* Cascading Style Sheets
* A language for styling HTML documents.
* It specifies certain rules for the layout and display of elements.
* It uses key/value pairs to indicate those rules.
* CSS is broken down into selectors and declarations:
  + Selector: defines the element(s) you want to style.
  + Declaration: The key/value pair defining the applied style.

CSS Box Model:

* CSS treats each HTML element in the document as a “box” Each box has its own 4 regions all that can be styled independently of each other.
* From inner to outer the regions are:
  + Content
  + Padding
  + Border
  + Margin
* Nice image here: <https://upload.wikimedia.org/wikipedia/commons/e/ed/Box-model.svg>

CSS locations:

* Three places that CSS can be declared for your HTML document:
  + External Style Sheet (This is a separate document that is linked into the HTML document.)
  + Internal in the head of the HTML document with the <style> tag.
  + Inline as an attribute of the actual element itself.
* Conflicts:
  + The CSS declared closest to the element will “win-out” overriding other CSS.
    - So inline will override styling in the head which will override styling in an external sheet.

CSS Properties:

* There are many, many properties that can be styled with CSS. Some include:
  + Border:
    - Border width
    - Border style (dashed, dotter, none etc)
    - Border color
  + Padding
    - Padding top/right/bottom/left.
    - Giving the generic padding with numbers will shorthand the filling out of these options:
    - 1 number = all four sides {padding:70px;}
    - 2 numbers = first number is the top and bottom, the second is right and left. {padding:35px 70px;}
    - 3 numbers = 1st the top, second is left and right, 3rd is the bottom {padding: 25px 45px 70px;}
    - 4 numbers = 1st is top, 2nd is right, 3rd is bottom, 4th is left. {padding: 25px 45px 70px 80px;}
  + Margin is similar to padding just on the other side of the border.
  + Display - determine the display behavior of the element. Inline or Block.
  + Position - defines where on the page the element is displayed.
    - Static - the element box is arranged normally.
    - Relative - defaults to the normal position but can change in relation to other elements.
    - Absolute - element’s box is placed in an absolute location on the page. (top-right for example)
    - Fixed - the element’s box is in a fixed location in the window, unaffected by user scrolling.
  + Color - allows you to change the color.
    - Color options including simple color names (red), rgb, hexadecimal, or hsl.
  + Text Align - sets the text inside the content to be of a particular alignment. i.e. left right justified or centered.

CSS Selectors:

* These are used in an external style sheet or the style tag to select single or multiple elements of the HTML document to style.
* Element Selectors: Select all instances of a particular type of element.
  + <style>

p {

text-align:center;

color:blue;

}

</style>

* ID selector: Selects a single element by it’s unique ID attribute. Uses the #.
  + <style>

#para1 {

text-align:left;

color:red;

}

</style>

* Class Selectors - will style any element with the declared class attribute. Uses the . to select. Can be combined with element selectors.
  + <style>

.intro {

text-align:center;

color:blue;

}

</style>

<style>

p.intro {

text-align:left;

color:red;

}

</style>

* Universal selector - selects everything. Uses the \*
  + <style>

\* {

font-family:arial;

color:pink;

}

</style>

* Attribute Selector - Allows you to select elements that declare a specific attribute. Uses the [] to select.
  + <style>

[title] {

text-align:left;

color:red;

}

[class=”warning”]{

color: green;

}

</style>

* Grouping Selectors: you can group multiple selectors and apply the same style. Simply declare the selectors and separate them with a ,
  + <style>

h1, #para1 {

text-align:left;

color:red;

}

</style>

* Child selectors - Selects specific elements that are direct children of another specific type of element. Uses the > sign.
  + <style>

div>p {

text-align:left;

color:red;

}

</style>

* Descendents selectors - like a child selector but will affect every element under the “parent” element no matter how far down it is nested. It uses an empty space character “ “ between the parent and the decedent.
  + <style>

div p {

text-align:left;

color:red;

}

</style>

* Sibling Selectors - selects an element that is a direct sibling to another element. Uses the ~ to select.
  + <style>

div~p {

text-align:left;

color:red;

}

</style>

* Adjacent Sibling Selectors - Same as sibling selectors but must immediately follow the other element in the HTML document. Uses the + to select.
  + <style>

div+p {

text-align:left;

color:red;

}

</style>

* Parents and siblings:
  + <div id=”mydiv”>
    - <p> This is a direct child of mydiv</p>
    - <div>
      * <p>a descendant of mydiv but not a direct child</p>
    - </div>
  + </div>
  + <p> This is adjacent sibling of mydiv</p>
* Conflicts - If selectors conflict with each other then the “winning” CSS is determined by specificity:
  + ID wins over everything.
  + Class beats anything beside ID.
  + Element is the lowest priority.
  + If specificity does not resolve the conflict the determination is made by the order of the declarations. The final declaration declared will win.
  + NOTE: Specificity is preferenced over the distance of the declaration.
* CSS3 is the current version on CSS. Its major feature updates were for mobile development.