Terms

CSS (Cascading Style Sheet): CSS defines how your HTML will look. Styles cascade according to the type of stylesheet and the order in which they are read by the browser. If there is more than one style declared for a tag the following takes precedence:

1. Inline styles
2. Internal or External stylesheets – whichever stylesheet is read last takes precedence
3. Browser default

Inline Styles: Inline styles are coded in the html tag using the style attribute. For example, <p style="color=#fff;">. Inline styles will only affect a single element.

Internal Stylesheet: An internal stylesheet is coded in the head of an HTML page using the <style> tag. Internal styles will only affect the elements of the HTML page.

External Stylesheet: An external stylesheet is a separate .css document that is linked to an HTML document in the head of the HTML page. An external stylesheet effects the elements of every HTML page that it is linked to.

Selector: What HTML element the CSS style is targeting. There are multiple types of selectors. This class discusses:

- **Element Selector:** Targets specified HTML tags. For example, p would style all the <p> tags.
- **Pseudo Selector:** Selectors that specify varying states of the element. For example, when :hover is added to the a element selector it will only apply the style when the mouse hovers over a link.

Declaration: The defined style coded as a property/value pair (property: value;)

Structure

Inline Structure

```
<p style="color:turquoise"> My paragraph will be turquoise</p>
```

property · value · declaration · HTML attribute

Internal Structure

```
<style>                    HTML tag that defines a CSS style. Declared in the head of the HTML document
    p {                    Selector that targets the HTML element to style
        color: red;        Declaration of property/value pair
        font-size: 16px;   Each property and value must be separated by a colon
                           Each declaration must be separated by a semicolon
    }
</style>
```

Nested declaration · Nested styles

External Structure

HTML: Add the following tag and attributes to the head of the HTML document

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

relationship attribute · type attribute · path to your CSS document · hyperlink reference attribute

CSS Document

```
@charset "UTF-8";          Declares the character type and document type.
/* CSS Document */         File must be saved as a .css document

p {                        Selector that targets the HTML element to style
    color: red;            Declaration of property/value pair
    font-size: 16px;       Each property and value must be separated by a colon
                           Each declaration must be separated by a semicolon
}
```

Nested styles

CSS Selectors discussed in class

Below is a brief synopsis of the new selectors discussed in class. For more information visit www.w3schools.com/cssref/css_selectors.asp

Description	Example
h1 selector. Targets the h1 HTML tag for styling.	`h1 { color: red; }`
p selector. Targets the p HTML tag for styling.	`p { color: red; }`
a selector. The a selector specifies the link element. Use pseudo selectors to specify styles for varying states of the element.	`a { color: red; }`
a:link pseudo selector. Specifies the style of a normal unvisited link.	`a:link { color: red; }`
a:visited pseudo selector. Specifies the style of visited link.	`a:visited { color: red; }`
a:hover pseudo selector. Specifies the style when the mouse hovers over the link.	`a:hover { color: lightred; }`
a:active pseudo selector. Specifies the style when the mouse is clicked.	`a:active { color: darkred; }`
Cascading order of pseudo selectors: The order of the styles is important. a:hover MUST come after a:link and a:visited. Then a:active MUST come after a:hover	
Grouping selectors. If you have multiple selectors with the same properties and values, you can group them together using a comma.	`a:link, a:visited { color: red; }`
CSS Comments. As with coding HTML documents you should use comments in your CSS stylesheets to help organize and explain what your stylesheet is doing.	`/* My comments */`
ul selector. The ul (unordered list) and ol (ordered list) selectors style how lists appear and which kind of markers if any are used to mark lists.	`ul { list-style-type: square; }`

ol selector. The ol (ordered list) selectors style how lists appear and which kind of markers if any are used to mark lists.

```
ol {
  list-style-type: lower-alpha;
}
```

table selector. As with coding HTML documents you should use comments in your CSS stylesheets to help organize and explain what your stylesheet is doing.

```
table {
  border-collapse: collapse;
}
```

th selector. As with coding HTML documents you should use comments in your CSS stylesheets to help organize and explain what your stylesheet is doing.

```
th {
  color: white;
}
```

td selector. As with coding HTML documents you should use comments in your CSS stylesheets to help organize and explain what your stylesheet is doing.

```
td {
  font-size: 20px;
}
```

tr selector. The tr selector specifies the style of the rows.

```
tr {
  font-size: 20px;
}
```

:nth-child() structural pseudo-selector. Style a piece of an element based on its order within the element. This is a structural pseudo-class that styles children of an element according to the relationship it has to its parent or siblings. For example, you could specify every odd numbered row by attaching: nth-child(odd) to the tr selector.

```
tr:nth-child(odd) {
  background-color: whitesmoke;
}
```

form selector. The form selector will style the area that contains the form elements.

```
form {
  background-color: whitesmoke;
}
```

Styling elements inside other elements. To specify styles that only effect the element that is inside another element separate the elements using a space.

```
form p {
  font-size: 16px;
}
```

input type selectors. There are many types of input selectors. To specify specific types, use the syntax below.

- **input** – styles all input selectors
- **input[type=text]** – styles only text fields
- **input[type=password]** – styles only passwords fields
- **input[type=number]** – styles only number fields
- **input[type=button]** – styles only buttons
- **input[type=submit]** – styles only submit button
- **input[type=reset]** – styles only reset button

```
input[type=text] {
  font-size: 12px;
  padding: 5px;
}
```

input[type=submit] selector. The input[type=submit] selector allows you to style the submit button.

```
input[type=submit] {
  color: white;
  border-radius: 10px;
  background-color: cornflowerblue;
}
```

button selector. The button selector lets you style buttons that use the button element.	`button { color: white; border-radius: 10px; background-color: cornflowerblue; }`
textarea selector. The textarea selector styles an input field for multiple lines of text.	`textarea { height: 150px; }`
:focus pseudo selector. The :focus pseudo selector declares styles for when the user has selected the input type and is typing in it or making a selection ect. It is helpful to style the :focus pseudo selector to help users keep track of where they are when filling out forms.	`form { background-color: whitesmoke; }`
footer selector. The footer selector lets you style elements that are in the footer container element.	`footer { background-color:cornflowerblue; color: white; }`

CSS Properties discussed in class

Below is a brief synopsis of the new properties discussed in class. For more information such as additional property values visit www.w3schools.com/cssref/default.asp. For a complete list of current **browser support** visit www.w3schools.com/cssref/css3_browsersupport.asp

Description	Example
color property. The color property refers to the color of text. Color values can be represented using a named value (red) or a Hexadecimal (#ff6347) as well as RGB, RGBA, HSL and HSLA color code.	`p { color: red; }`
font-family property. The font-family property refers to the type of font to use for the text. Fonts can be defined in 2 ways – a generic family such as serif vs non-serif or as a specific font family such as Arial or Times New Roman.	`p { font-family: Arial, 'sans-serif'; }`
font-size property. The font-size property specifies how large the text appears. Font-size values can be set using pixels, em, and others.	`p { font-size: 20px; }`
text-decoration property. The text decoration property specifies added decorative elements –underlines, overlines, linethrough.	`p { text-decoration: underline; }`
list-style-type property. The list-style-type property styles the type of bullet or numerical/alphabetical marker that you want to use for unordered or ordered lists.	`ul { list-style-type: circle; }`

width property. Width can be specified using measurements such as px or percentage of the page or container. You can use auto as a value to have the browser calculate the value for you.	`table {`
	`  width: 600px;`
	`}`

height property. Width can be specified using measurements such as px or percentage of the page or container. You can use auto as a value to have the browser calculate the value for you.	`table {`
	`  height: 300px;`
	`}`

background-color property. The background-color property sets the colour of the background of an element. As with the color property the color value can be specified using a named value (red) or a Hexadecimal (#ff6347) as well as RGB, RGBA, HSL and HSLA color code.	`table {`
	`  background-color: whitesmoke;`
	`}`

border property. Border styles the width, style (solid, dotted ect) and color of a container or table. It is shorthand for border-width, border-style, border-color.	`td {`
	`  border: 1px solid lightgrey;`
	`}`

border-radius property. The border-radius property defines the curve of the corners of a container or button. The value is specified using px.	`table {`
	`  border-radius: 6px;`
	`}`

Outline property. The outline property defines a line outside a border. This property is often used to highlight or make an element standout.	`input[type=text]:focus {`
	`  border: 2px solid cornflowerblue;`
	`  outline: red;`
	`}`

border-collapse property. Border-collapse is specific to table borders. Tables can have a border around each cell and the table, making it appear that the table has 2 borders. To merge multiple borders use the border-collapse property.	`table {`
	`  border-collapse: collapse;`
	`}`

margin property. Margin specifies the space outside a container. The margin short hand specifies the values in clockwise order starting at the top. The measurement can be length such as px, auto or percentage.	`th {`
	`  margin: 5px 10px 5px 10px;`
	`}`

padding property. Padding specifies the space inside a container. The padding short hand specifies the values in clockwise order starting at the top. The measurement can be length such as px, auto or percentage.	`form {`
	`  padding: 20px 50px;`
	`}`

text-align property. The text-align property specifies how text is aligned horizontally in a container.	`td {`
	`  text-align: center;`
	`}`

Explore on your own

Below is a list of additional property values you may want to explore.

Description	Example

Color Values:
→ Hexadecimal colors: #ff0000;
→ RGB colors: rgb(red, green, blue)
→ RGBA colors: rgba(red, green, blue, alpha)
→ HSL colors: hsl(hue, saturation, lightness)
→ HSLA colors: hsla(hue, saturation, lightness, alpha)
→ Predefined/Cross-browser color names: red
 	
To explore these color values, go to www.w3schools.com/cssref/css_colors_legal.asp.
	`p {`
	`  color: #ff0000;`
	`}`

Units of Measurement - absolute values:
→ cm
→ mm
→ in
→ px - pixels are relative to the viewing device
→ pt - points (1pt = 1/72 of 1in)
→ pc - picas (1pc = 12 pt)
 	
To explore these units of measurement, go to www.w3schools.com/cssref/css_units.asp
	`table {`
	`  margin: 5px 10px 5px 10px;`
	`}`