

Logols Learning

WEEKEND WEB DEVELOPMENT BOOT CAMP

TRAINING: CSS

What is CSS?

- ▶ HTML provides the definition of the page
- ▶ CSS provides the styling for the page
- ▶ It's best to keep these separate
- ▶ Cascading Style Sheets
- ▶ If multiple styles applied, styles will override based on the hierarchy

- CSS provides the styling for the page.
- It stands for cascading style sheets

Inline Styling

- ▶ Can style inside an element using the style attribute
- ▶ This overrides all other styling
- ▶ Style Attribute is a global attribute
- ▶ Can be used on any html element
- ▶ `style="[Style Name1]: [Value]; [Style Name2]: [Value];"`

Example:

```

```

- The styles can be specified in multiple places.
- First you can specify them in the style attribute of an element.
- The style element is global and can be used on any html element.
- The CSS styles are specified by using the style name followed by : followed by the value.

In Page Styling

- ▶ It's best to separate your elements from the style
- ▶ You can also style in the page
- ▶ Use style element within the head element
- ▶ Surround styles in a comment

Example:

```
<head>
<style>
<!--
    .logo {
        height:100px;
        width:100px;
    }
-->
</style>
</head>
...

```

- You can also specify styling within the page.
- Add the style element within the head element
- Then styles can be specified.

Linking a CSS File

- ▶ It's best to style outside of the html file altogether in a .css file
- ▶ To link use the link element inside the head
 - ▶ Attributes:
 - ▶ rel - stylesheet
 - ▶ href – location of css file
 - ▶ type – text/css

Example:

```
<head>
  <link rel="stylesheet" href="/styles/site.css" type="text/css" />
</head>
```

- Finally you can also specify styles in another file.
- This is the recommended approach.
- Link the css file by adding a link element to the head element
- Specify rel as stylesheet
- Specify type as text/css
- Then specify the href as the location of the css file.

Selectors

- ▶ Used to select elements for styling
- ▶ Sample below

Name	Syntax	Example
ID	# [Element ID]	#PersonTable
Class	. [Class Name]	.highlightTable
All Elements	*	*
Element	[Element]	p
Element in Element	[Element] [Element]	div p
Multiple Elements	[Element], [Element]	div, p
Element direct child	[Element] > [Element]	div > p

- These are different selectors that can be used to set a style for
- The most obvious ones are ID, class, and element
- To specify an id, put a # in front of it.
- To specify a class, put a . In front of it.
- To specify an element, just write it.
- With these selectors it could apply to many elements throughout the site
- All of these elements are updated with the style.

Comments

- ▶ Comments should be surrounded by /* */

Single line Example:

```
/* this is a comment*/
```

Multiple Line Example:

```
/*
This is also
a comment
*/
```

- Comments are provided with the /* */ syntax.
- Single and multi line comments are the same.

Colors

- ▶ Uses: font, background, border
 - ▶ color: red;
 - ▶ background-color: red;
 - ▶ border-color: red;
- ▶ Multiple ways to specify colors:
 - ▶ Color Name
 - ▶ RGB
 - ▶ Hex

Example:

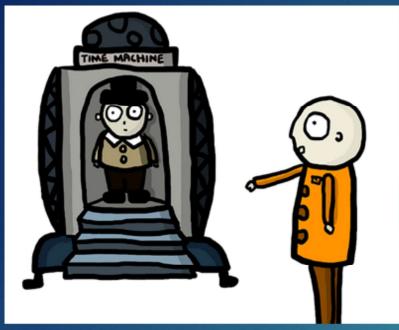
```
color: red;  
color: #00ff00;  
color: rgb(0,0,255);
```

- You can set the font color, background color, or border color.
- The font color is specified using the color style
- The background color is specified using the background-color style
- The border color is specified using the border-color style
- Colors can be specified in multiple ways.
- There are many colors that are named.
- In this way you can set the color by its name
- You can also use RGB, which is specifying the values for red, green, and blue.
- Finally, you can use the hex value.

Text

- ▶ color - Example: color: red;
- ▶ text-align – Example: text-align: center;
- ▶ text-decoration – Example: text-decoration: underline;
- ▶ text-transform – Example: text-transform: lowercase;
- ▶ letter-spacing – Example: letter-spacing: 5px;
- ▶ text-indentation – Example: text-indent: 25px;
- ▶ line-height – Example: line-height: 1;
- ▶ word-spacing – Example: word-spacing: 10px;
- ▶ text-shadow – Example: 5px 5px red;
- ▶ Direction – Example: direction: rtl;

- There are many styles for text.
- As just mentioned there is color for the text color.
- There is text-align to specify the horizontal alignment.
- And much more



EXAMPLE

LINKING, SELECTORS, COLORS, TEXT, COMMENTS

ASSESSMENT

LINKING, SELECTORS, COLORS, TEXT, COMMENTS



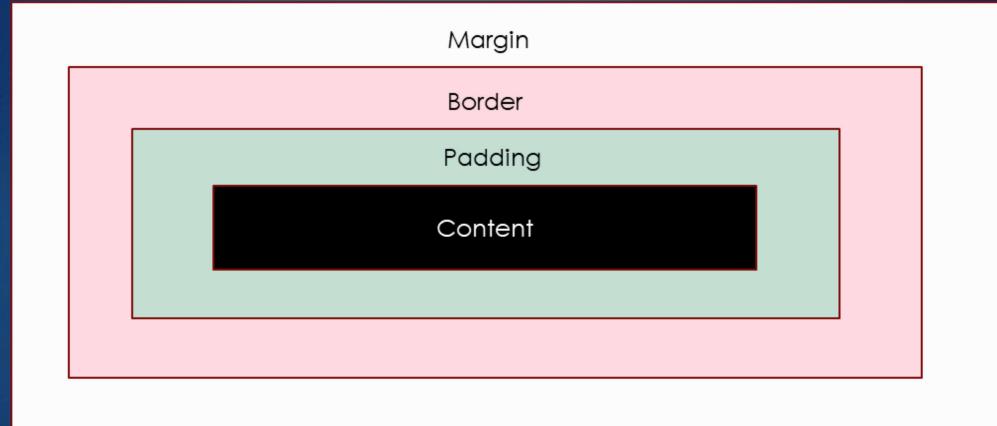
- What is CSS used for?
- What does CSS stand for?
- What tag is used to reference a separate CSS file?
- Write on the board the css to make the font color red for an element with an id of header.

Assignment

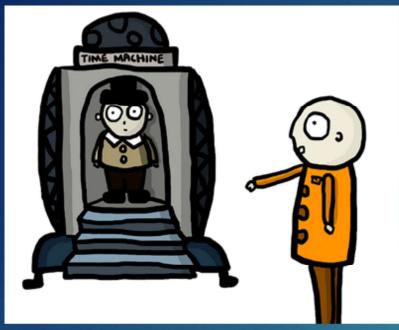
- ▶ Style the Status Report Interface.
- ▶ Center the header text using an id selector.
- ▶ Modify the paragraph text to be dark grey using a class selector.
- ▶ Modify the table header to be light grey using the element selector.



Box Model



- The box model is an important part of laying out parts of a page.
- The content is surrounded by the padding. The padding is surrounded by the border. The border is surrounded by the margin.
- All of these need to be considered when figuring out the total size of the content.



EXAMPLE

BOX MODEL

Display

- ▶ Options:
 - ▶ inline
 - ▶ block
 - ▶ flex
 - ▶ inline-block
 - ▶ inline-flex
 - ▶ table
 - ▶ none
 - ▶ etc...
- ▶ Display:none vs. Visibility
- ▶ Display Examples:
<https://css-tricks.com/almanac/properties/d/display/>
- ▶ Flexbox Examples:
<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

- Display has a number of options.
- One of those is display none.
- Display none means the element will not appear on the page
- This is different than visibility which will make it not show on the page, but still show in the DOM.
- Let's review examples of display and flexbox.

Position

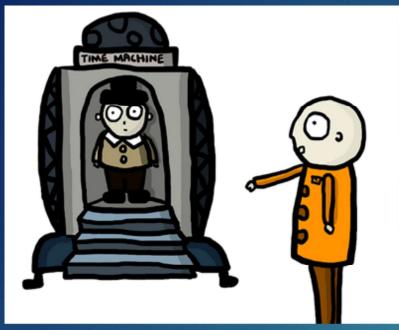
- ▶ Options:
 - ▶ static
 - ▶ relative
 - ▶ fixed
 - ▶ absolute
- ▶ Examples: <http://learnlayout.com/position.html>

- Position is another style to change how an element is positioned on a page.
- Let's review some examples.

Float

- ▶ float
 - ▶ none
 - ▶ left
 - ▶ right
 - ▶ initial
 - ▶ inherit
- ▶ clear
 - ▶ Same +both
- ▶ Example: <https://css-tricks.com/all-about-floats/>
- ▶ overflow
 - ▶ visible
 - ▶ hidden
 - ▶ scroll
 - ▶ auto
 - ▶ initial
 - ▶ Inherit
- ▶ Example: <https://css-tricks.com/almanac/properties/o/overflow/>

- Here are some other styles that effect the position of elements.
- It's easier to see these in action again, so let's look through the examples.



EXAMPLE

DISPLAY, POSITION, FLOAT, FLEX BOX

Assignment

- ▶ Style the report so the image is to the right of the data table using floats.



Alignment

- ▶ Center Align Block Element or image
 - ▶ margin: auto;
- ▶ Center Align Text
 - ▶ text-align: center;
- ▶ Absolute Position Alignment
 - ▶ position: absolute
 - ▶ right: 0px; or left: 0px;
- ▶ Float Alignment
 - ▶ float: right; or float: left;
- ▶ Center Vertically with Padding
 - ▶ padding: 70px 0;
- ▶ Example:
https://www.w3schools.com/css/css_align.asp

- Here are some different ways to align elements.
- Let's again go through some examples.

Pseudo-Class

- ▶ Allow to set style of element in certain states.
- ▶ Examples of Pseudo-Classes:
 - ▶ a:active
 - ▶ a:hover
 - ▶ a:link
 - ▶ a:visited
- ▶ input:checked
- ▶ input:disabled
- ▶ input:enabled
- ▶ input:focus
- ▶ input:invalid
- ▶ input:valid

- CSS also has pseudo-classes.
- These allow different styles for different states of an element.
- Here are some examples of pseudo-classes
- For example pseudo-classes for a
 - a:active
 - a:hover
 - a:link
 - a:visited

Where is my style?

- ▶ Importance
 - ▶ Add !important to the end of a property
 - ▶ Wins over all other rules (not recommended)
- ▶ Specificity
 - ▶ Measured using 4 values
 - ▶ Thousands – inside style element or style attribute
 - ▶ Hundreds – one for each ID selector
 - ▶ Tens – one for each class selector, attribute selector, or pseudo-class
 - ▶ Ones – one for each element selector or pseudo-element
- ▶ Source Order
 - ▶ Later rule wins if everything else is the same

- This refers to the hierarchy of styles.
- Every property figures out what styles should be used, since conflicts will come up.
- The first check is based upon whether it is specified with !important at the end of the property.
- This wins over all other properties unless both are specified as !important
- After that specificity is used to determine the styles used.
- Specificity is determined by a score with 4 digits.
- The thousands place gets a value if its inside the style element or style attribute
- The hundreds place gets a value for each id selector
- The tens place gets a value for each class selector, attribute selector, or pseudo-class used.
- The ones place get a value for each element selector or pseudo-element
- Finally if the specificity is the same the later style wins.

Page Layouts

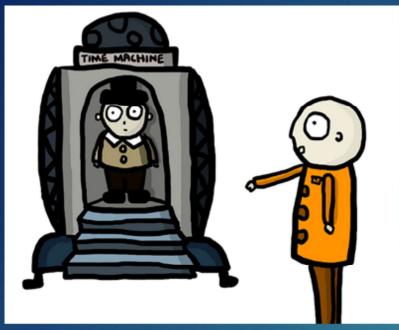
- ▶ Options:
 - ▶ Tables
 - ▶ Float
 - ▶ CSS Framework
 - ▶ Flexbox
 - ▶ CSS Grid
- ▶ Example:
https://www.codementor.io/code_mentorteam/4-different-html-css-layout-techniques-to-create-a-site-85l9t1x34
- ▶ Bootstrap Templates:
https://www.w3schools.com/bootstrap/bootstrap_templates.asp
- ▶ Scroll Template:
https://www.w3schools.com/bootstrap/bootstrap_theme_company.asp

- Page layouts are created using the same styles we've talked about so far.
- Tables can be used and was really popular for a while, but is not currently recommended.
- Floats can also be used to style the page layout.
- A framework can also be used to aid the creation of a layout
- Flexbox is newer and can be used, but a lot of browsers still in use do not support it
- Let's go through some examples.

Responsive

- ▶ Designing to work on any monitor size or device
- ▶ Use a Grid System – Bootstrap
- ▶ Use Media Queries
- ▶ Use Vector Graphics or Alternate Images
- ▶ Use a Framework like Bootstrap

- Pages need to be designed to work on different devices and screen sizes.
- These are some things that you can use to help.
- We are going to focus on using the bootstrap framework.

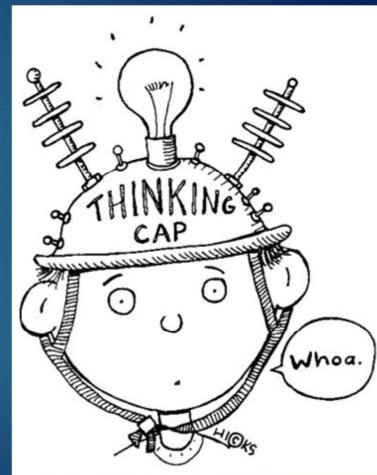


EXAMPLE

CSS GRID

ASSESSMENT

BOX MODEL, DISPLAY, POSITION, FLOAT, RESPONSIVE



- What makes up the box model?
- Write on the board the css to set the display to inline for a class named inlineText.
- Write on the board the css to fix the footer element to the bottom of the page.
- What are some options for page layout?
- What can be done to make a page responsive?

Assignment

- ▶ Add a div surrounding everything under the header.
- ▶ Add a max width to the div.
- ▶ Center the div using margin: 0 auto;
- ▶ Play around with margins and padding to change the looks.



QUICK REVIEW

CSS



- What tag is used to reference a separate CSS file?
- Write on the board the css to make the font color red for an element with an id of header.
- What makes up the box model?
- Write on the board the css to set the display to inline for a class named inlineText.
- Write on the board the css to fix the footer element to the bottom of the page.

Additional Resources

- ▶ JSFiddle
 - ▶ <https://jsfiddle.net/>
- ▶ free Code Camp
 - ▶ <https://www.freecodecamp.org/>
- ▶ Can I Use
 - ▶ <https://caniuse.com/>