Udemy CSS notes

# Section 4: Intro to CSS

CSS is a stylesheet code.

It’s best to use an external stylesheet and then add a link to your main HTML page

     <link rel=“stylesheet” href=“css/styles.css”>

HTML and CSS are just a bunch of boxes. CSS allows for flexibility in positioning boxes. Each box is a “div”.

Graphical user interface, text, application

Description automatically generated

The browser has defaults that sometimes need to be overridden in CSS.

Margins, border, padding, (object?)(selector?). all together, they are the DOM?

Position is a static definition of where something physically sits on a page in relation to the entire page. If you set an object to absolute, it will put each object to the default position of any first item on a page. After that, you can set position parameters to manually adjust relationships by eye.

Display will answer to position first, but it has settings to adjust to the objects before it.

[Display](https://www.w3schools.com/cssref/pr_class_display.php#gsc.tab=0): none, block, inline, inline-block, contents, and there are many more

# Section 5: Intermediate CSS

Typography: You can use px, %, em, rem, pt. I couldn’t figure out why my font size wasn’t matching. It turned out that I had Chrome font size set to “Large” when the default is “Medium”.

📌 One of the lessons included making two div sections 50% of the page width using a class (.skill-row). The challenge included adding a width value of 50% and adding a margin of auto. But I found success with changing the display value to inline-block. I’m not sure what the difference is or if one is better than the other (presumably Angela’s is foolproof).

float: We applied this to an image using a class. It floats inside the container, and all other elements in that container flow around it. We positioned one image to float left and another the float right. It modifies the display value.

clear: When you want any of these elements to NOT wrap around the floated item, use this and it will go below (or above??) the float and wrap elements.

Only use float for wrapping text, you will get in trouble using it for anything else.

For borders, make sure to always have the border style in the code - in this case, solid:

  border: 2px solid black;

Button generator:

<https://css3buttongenerator.com/>

Front-end Mentor:<https://www.frontendmentor.io/challenges/news-homepage-H6SWTa1MFl/hub/news-homepage-GBHBmze9Yp>

Front-end Mentor: getting the most out of it  <https://medium.com/frontend-mentor/how-to-get-the-most-out-of-frontend-mentor-bdd6fdc25cb8>

Article: <https://www.joshwcomeau.com/css/pixel-perfection/>

# Sections 6 and 7 - Bootstrap

Carousel

Pages change horizontally. Options are automatic, with buttons manually, with indicators to show which window you are on, and combinations.

Carousels always need id to work.

z-index: determines how the elements are layered (back to front). Default is lower is on top.

Media query: responds to different types of output

@media <type> <feature>

## Code refactoring

1. Readability
2. Modularity
3. Efficiency
4. Length

**Combining selectors**

HTML Elements

element1, element2 - comma separated, reads RTL, applies to each element in HTML (space optional - h1,h2)

h1, h2 {

color: red;

}

-------🡪

<h1>Hello</h1> Hello

<h2>Hello</h2> Hello

Hierarchical

selector1 selector2 - one space, parent followed by child, reads RTL. In example, #title is parent, .container-fluid is child. It will look for any selector .container-fluid is inside the ID #title.

#title .container-fluid {

color: red:

}

-------🡪

<div id=”title”>

<div class=”title”>Hello</div> **Hello**

<div class=”container-fluid”>Hello</div> **Hello**

<div>

Combined selectors

Selector1selector2 - no separator, reads LTR. Effects only an element with both classes.

#title.container-fluid {

color: red;

}

-------🡪

<div id=”title”>

<div class=”title”>Hello</div> **Hello**

<div class=”container-fluid”>Hello</div> **Hello**

<div class=”title container-fluid”>Hello</div> **Hello**

<div>

NOTE:

Unless applying Bootstrap, do not use multiple classes in an element. It’s bad practice because it can be confusing and cause problems.

Try not to add styling that can be done in CSS or Bootstrap directly to HTML code.