HTML & CSS Part 1

=> HTML basics
=> DOM
=> CSS Basics
=> Advance selectors
=> Pseudo classes
=> pseudo elements
=> absolute units
=> responsive units
=> CSS Browser support

1. What is HTML

*“HTML (**H**yper**t**ext **M**arkup **L**anguage) is the code that is used to structure a web page and its content.”*

Document Structure

```html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>My test page</title>
  </head>
  <body>
    <img src="images/firefox-icon.png" alt="My test image">
  </body>
</html>

```

<!DOCTYPE html> - declaration represents the document type, and helps browsers to display web pages correctly.

<html></html> - root element, all elements should be nested.

<head></head>

“This element acts as a container for all the stuff you want to include on the HTML page that isn't the content you are showing to your page's viewers. This includes things like keywords and a page description that you want to appear in search results, CSS to style our content, character set declarations and more.”

- MDN Web docs

<meta charset="utf-8"> - This element sets the character set your document should use to UTF-8 which includes most characters from the vast majority of written languages.

<title></title> — the <title> element. This sets the title of your page, which is the title that appears in the browser tab the page is loaded in.

<body></body> — the <body> element. This contains *all* the content that you want to show to web users when they visit your page.

Tag structure

Elements can have attributes

Self-closing tags

```
1 <img src="./kitten.jpg" alt="A kitten"/>
2
3 <input type="text" placeholder="Provide your name"/>
```

* http://xahlee.info/js/html5_non-closing_tag.html

HTML TAGS

- <header>
-
- <footer>
- <p>
- <h1> to <h6>
-
-
-
- <form>
- <input>
- <button>
- <select>
- <option>
- <label>
- <a>
-

* https://www.w3schools.com/tags/ref_byfunc.asp

`a` and `img` attributes

<a>

```
href=""
href="mailto:"
href="tel:"
target="_blank"
download
title=""
```



```
src=""
srcset="small.jpg 320w,
        medium.jpg 768w"
sizes="(max-width: 320px)
280px, (max-width: 768px)
740px, 1000px"
alt=""
```

Forms: <input/> types

Most used:

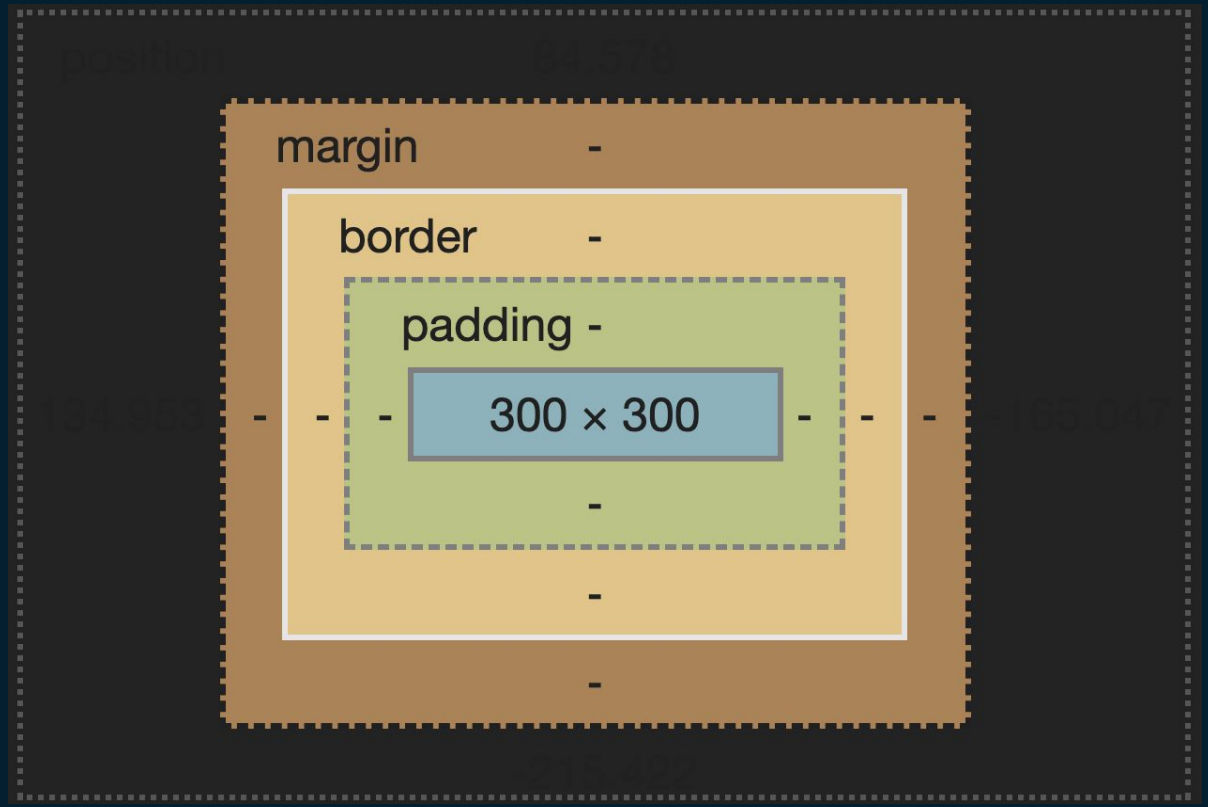
- “text”
- “number”
- “password”
- “email”
- “checkbox”
- “radio”

Good to know:

- “date”
- “color”
- “file”
- “hidden”

Attributes: https://www.w3schools.com/tags/tag_input.asp

BOX MODEL



Block and inline elements

This is "p" tag!

This is "p" tag!

This is span 1 This is span 2

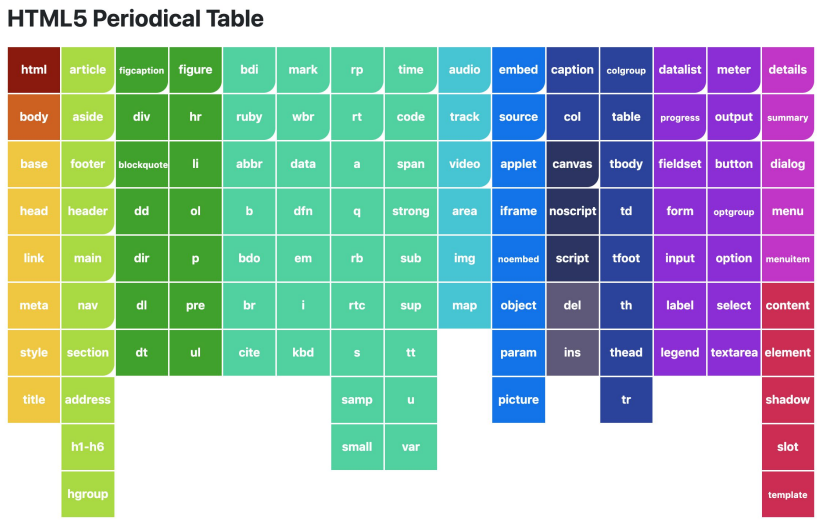
* https://www.w3schools.com/html/html_blocks.asp

Inline elements limitations

To inline elements you can’t apply the next styles:

- height;
- width;
- Margin-top and margin-bottom(only left and right)

HTML5 Periodic Table

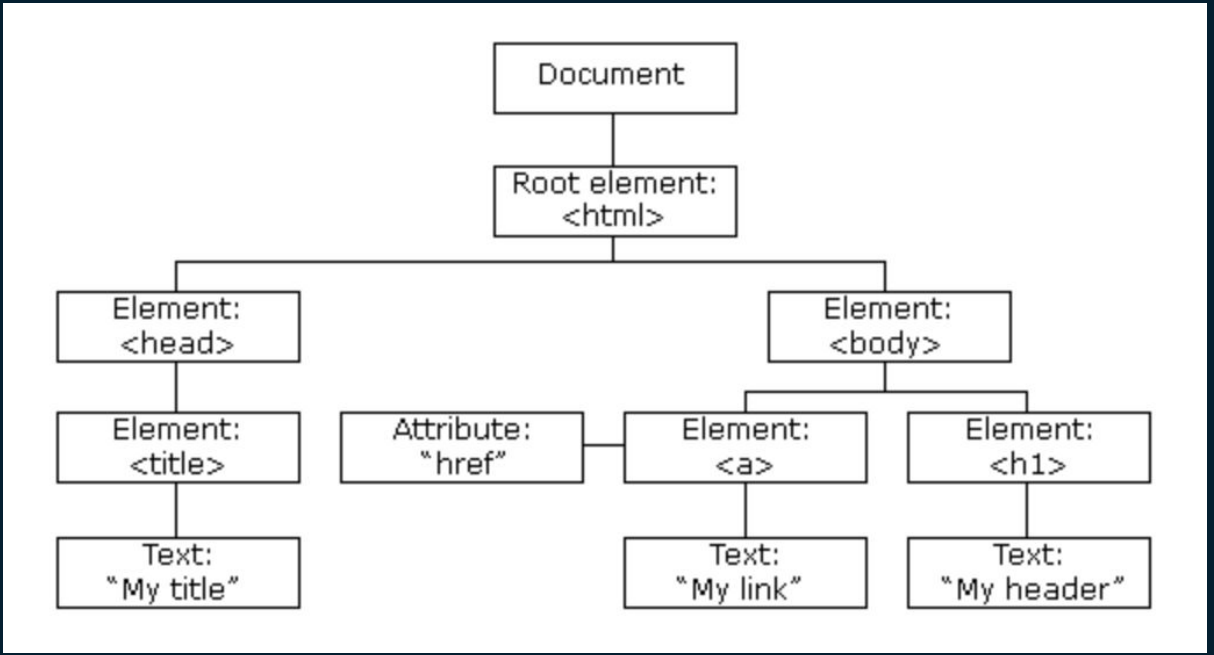


https://websitesetup.org/html5-periodical-table/

2. DOM

DOM ***- is a document object model***

- **A browser creates DOM** - takes your HTML and create a kind of duplicated structure(might fix something for you).
- **HTML elements as an object** which will get some extra properties.
- **Properties of all HTML element**(getElementById)
- **Events for elements**(mouseenter, click etc)
- **Methods to access all HTML elements**(innerText, InnerHTML, classList, etc)

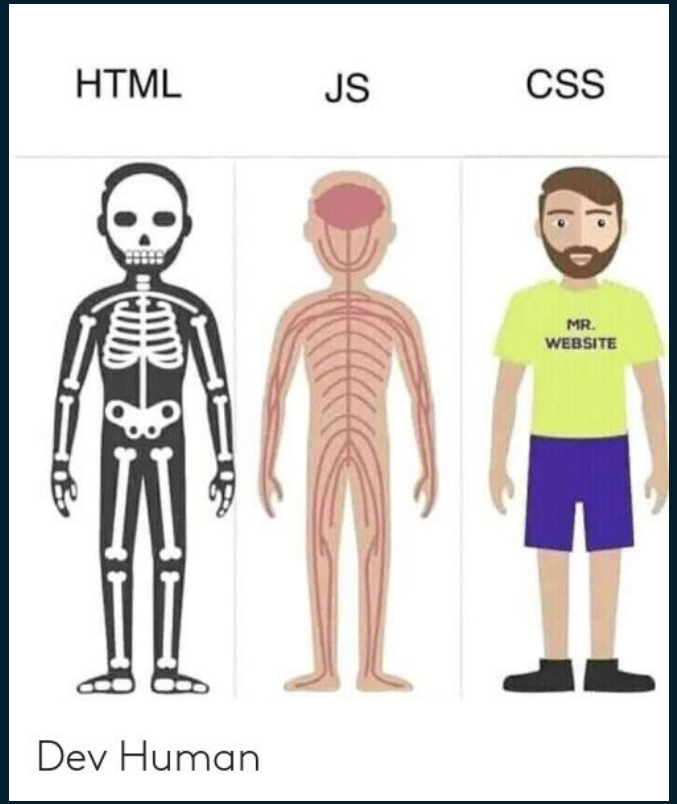


* Source W3School

A good article to read - https://css-tricks.com/dom/

3. What is CSS

CSS - Cascading style sheets



Ways to add styles:

- Inline(high priority, messy code)
- Head styles(cleaner and saves loading time)
- CSS file - (separation of concerns)

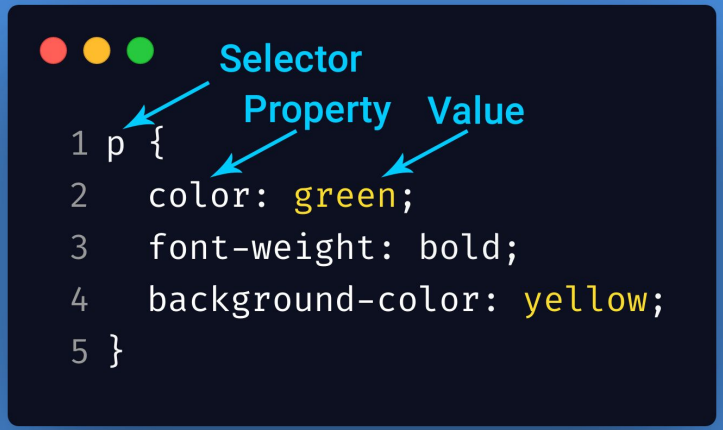
Inline styles

```
1 <h1 style="color: green;">For those who still with us</h1>
```

Head styles

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Coders In Hood </title>
  <style>
    h1 {
     color: green;
    }
  </style>
</head>
```

Basic structure



Specificity

=> Tag name
=> Class
=> Id
=> * **!Important** value - redefines everything

Calculator - *https://specificity.keegan.st/*


When your css property doesn't work
!important

A few extra bits

=> Combining selectors(h1, h2, h3...)

=> color systems

=> css units - *http://www.w3schools.com/cssref/css_units.asp*

Advance Selectors

We already know

=> Tag

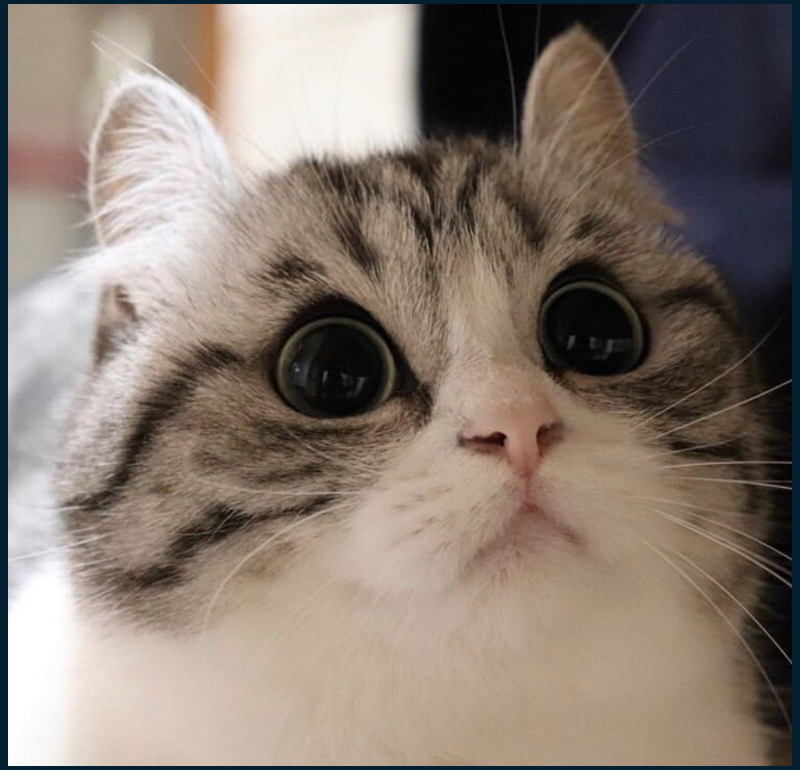
=> Class

=> Id

We have more ways to select

=> tag/class/id concatenation(.btn.btn_red)
=> Nested elements (.list .list_item)
=> Select a few elements (.list_1, .list_2)
=> “div **>** p” – select all p element with direct “div” parent
=> “div **+** p” – <p> elements that are placed immediately after elements(first sibling)
=> “div **~** p” – select all “p” that are preceded by a “div” element

… and even more



Attribute selectors(1)

=> [placeholder=”First name”] - select all elements where attribute “placeholder” = “First Name”
=> [placeholder~=”name”] - select all elements where attribute “placeholder” contains “name”
=> [placeholder|=”name”] - select all elements where attribute “placeholder” starting with “name”(divider must be “–”, ex. “name-last”)

Attribute selectors(2)

=> [placeholder^=”name”] - select all elements where attribute “placeholder” begins with “name”
=> a[href$=”.pdf”] - selects every <a> element where href attribute value ends with ".pdf"
=> a[href*=”codersinhoods”] - selects every <a> element where href attribute contains substring “codersinhoods”

Pseudo-classes

“A CSS pseudo-class is a keyword added to a selector that specifies a special state of the selected element(s).”

– MDN web docs

Most popular

:active
:checked
:disabled
:first-child
:last-child
:nth-child()

:focus
:has()
:hover
:not()
:visited

https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-classes

Pseudo-elements

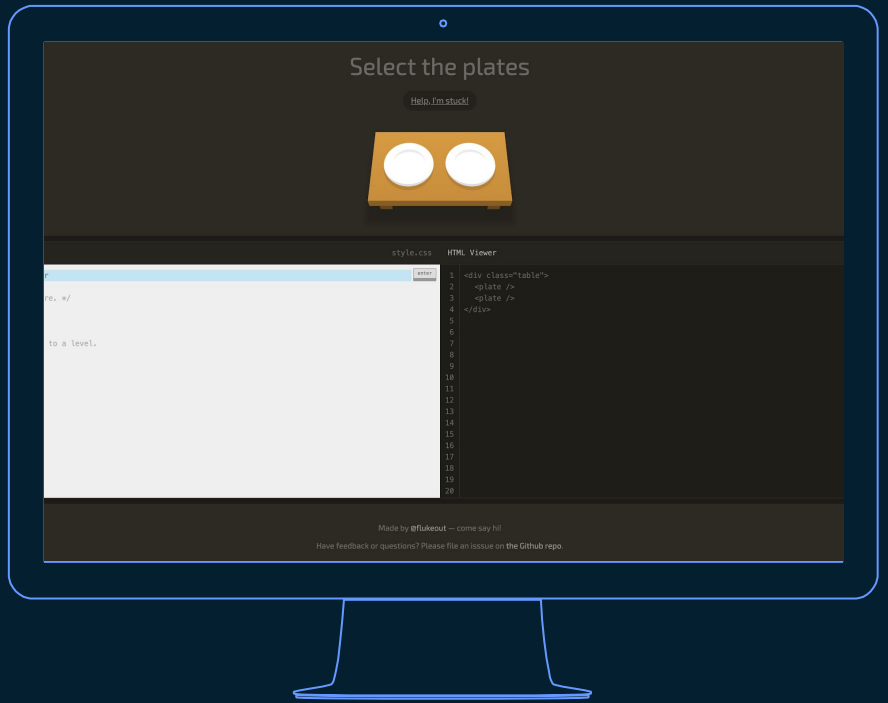
“A CSS pseudo-element is used to style specified parts of an element.”

– MDN web docs

::after
::before
::first-letter
::first-line
::placeholder

GAME TO PRACTICE

https://flukeout.github.io/



Absolute units

=> cm
=> mm
=> in
=> px
=> pt - points
=> pc - picas

Responsive units

=> em - Relative to the font-size of the element (2em means 2 times the size of the current font)
=> ex - Relative to the x-height of the current font (rarely used)
=> ch - The ch unit sets the font-size relative to the width of the character "0".
=> rem - Relative to font-size of the root element
=> vw - Relative to 1% of the width of the viewport
=> vh - Relative to 1% of the height of the viewport
=> vmin - Relative to 1% of viewport smaller dimension
=> vmax - Relative to 1% of viewport larger dimension
=> % - Relative to the parent element

Browser support

https://caniuse.com/

Good to read

https://www.cssbasics.com/introduction-to-css/