CSS crash course

1. Before we begin

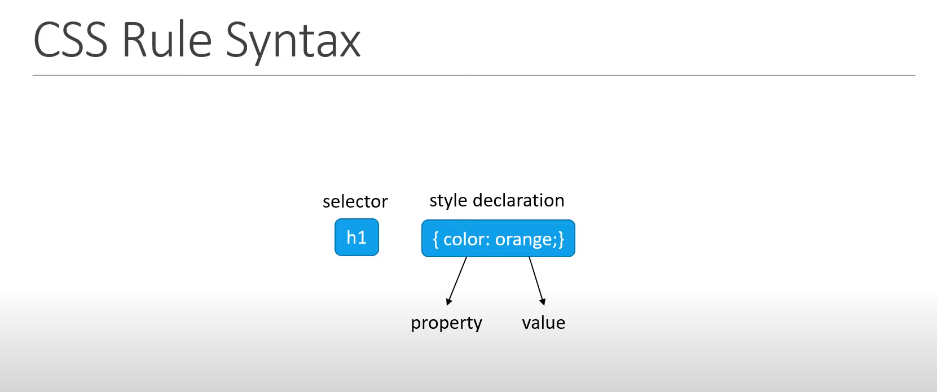
* This course iss for complete beginners
* Continuation of the HTML Crash Course
* We’re gonna learn the fundalmentals of CSS

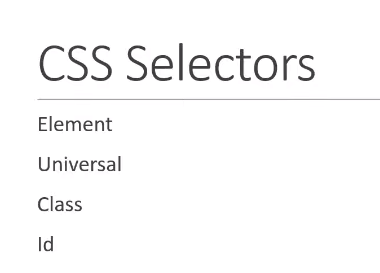
\*) support tools:  
- Visual code extension: Prettier – Code formatter  
- page ruler chrome extension

1. What is CSS?

* Language used to style an HTML document
* It’s a stylesheet language, not a programming language
* Select HTML elements and apply styles

1. How to add CSS?
2. Inline styles
3. Internal stylesheet
4. External stylesheet



1. CSS selectors  
   - Basic CSS selectors:  
   

* Full CSS selectors:

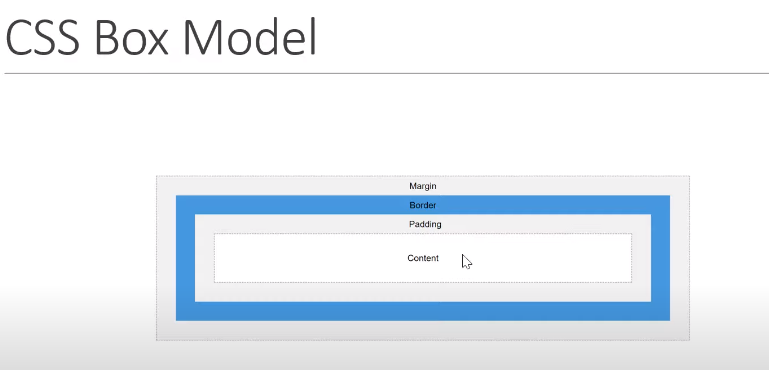
CSS selectors are patterns used to select and style HTML elements based on their attributes, types, classes, IDs, and relationships with other elements. Understanding CSS selectors is fundamental to effectively styling web pages. Here are some key points about CSS selectors:

1. **Element Selector**: Selects HTML elements based on their element type. For example, **p** selects all **<p>** elements.
2. **Class Selector**: Selects elements with a specific class attribute. It is denoted by a dot (**.**) followed by the class name. For example, **.my-class** selects all elements with **class="my-class"**.
3. **ID Selector**: Selects a single element with a specific ID attribute. It is denoted by a hash (**#**) followed by the ID name. For example, **#my-id** selects the element with **id="my-id"**.
4. **Attribute Selector**: Selects elements based on their attributes and attribute values. It can be used to select elements with specific attributes or attribute values. For example, **[type="text"]** selects all elements with **type="text"**.
5. **Descendant Selector**: Selects an element that is a descendant of another element. It is denoted by a space () between two selectors. For example, **div p** selects all **<p>** elements that are descendants of **<div>** elements.
6. **Child Selector**: Selects an element that is a direct child of another element. It is denoted by a greater than sign (**>**) between two selectors. For example, **div > p** selects all **<p>** elements that are direct children of **<div>** elements.
7. **Adjacent Sibling Selector**: Selects an element that is immediately preceded by another element. It is denoted by a plus sign (**+**) between two selectors. For example, **h2 + p** selects all **<p>** elements that are immediately preceded by an **<h2>** element.
8. **General Sibling Selector**: Selects elements that are siblings of a specified element. It is denoted by a tilde (**~**) between two selectors. For example, **h2 ~ p** selects all **<p>** elements that are siblings of an **<h2>** element.
9. **Pseudo-classes**: Selects elements based on their state or position. Common pseudo-classes include **:hover**, **:active**, **:focus**, **:first-child**, **:last-child**, and **:nth-child()**. For example, **a:hover** selects all **<a>** elements when they are being hovered over.
10. **Pseudo-elements**: Selects parts of an element that are not represented by regular HTML elements. Common pseudo-elements include **::before**, **::after**, **::first-line**, and **::first-letter**. For example, **p::first-line** selects the first line of every **<p>** element.
11. Common Styling Properties

* color & background-color
* Font:
  + font-family
  + font-size
  + font-weight

1. CSS box model

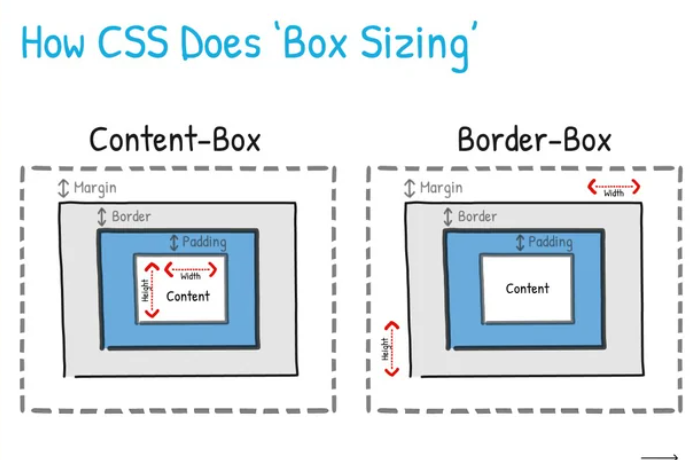
* In web design, every element is considered a rectangular box
* The CSS box model is sort of a standard by which browsers render HTML elements
* The box model consists
  + Margins: the space outside the border of the element, separate it from other elements
  + Borders: the border surround the content and padding of the element.
  + Padding: the space between the content and the border of the element. Padding adds internal space within the element
  + Actual content: text or image. The content area is determined by the width&height props



* The total width and height of an element:

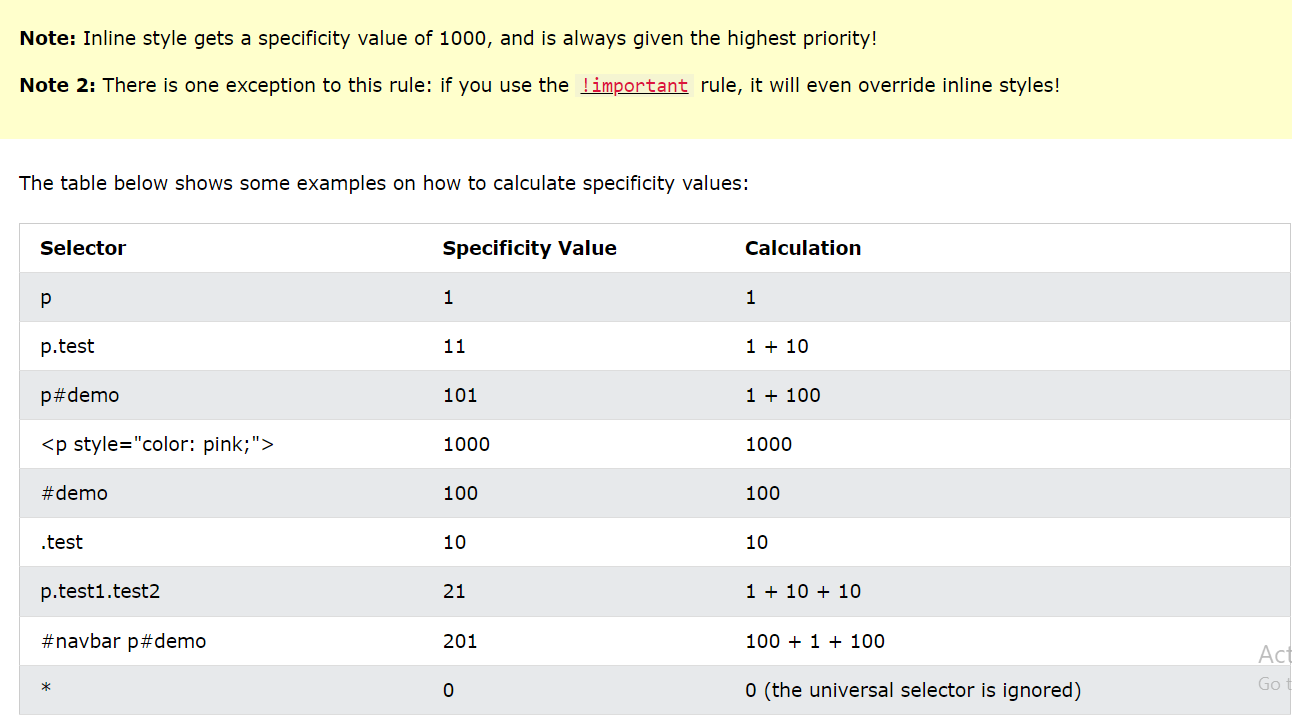
|  |
| --- |
| Total Width = Width + Left Padding + Right Padding + Left Border + Right Border + Left Margin + Right Margin  Total Height = Height + Top Padding + Bottom Padding + Top Border + Bottom Border + Top Margin + Bottom Margin |

* It's important to note that the CSS Box Model can sometimes cause unexpected layout behavior, especially when dealing with padding, border, and margin. To avoid issues, you can use the **box-sizing** property to control how the total width and height of an element are calculated.
  + box-sizing: content-box (default) 🡪 which includes only the content area in the width and height calculations.
  + box-sizing: border-box 🡪 includes padding and border in the calucations, which can make it easier to create consistent layouts.



1. Cascade, specificity & inheritance

* Cascading: when 2 rules apply that have equal specificity, the one that comes last in the CSS is the one that will be used.
* Specificity: is how the browser decides which rule applies if multiple rules have different selectors but apply to the same element.



* Inheritance: some css property values set on parent elements are inherited by their child elements, and some aren’t.

VD: color for text

* + Shouldn’t use this

1. Display

* block | inline | none | flex | grid

1. Position

* static | relative | absolute | fixed
  + static: Positioned statically. It's the default position value, and elements with this position value are positioned according to the normal flow of the document.
  + relative: Positioned relative to its normal position. By adding **top** and **left** properties, you can shift it from its original position.
  + absolute: Positioned relative to the nearest positioned ancestor (if any). If there's no positioned ancestor, it's positioned relative to the initial containing block (usually the viewport).
  + fixed: Positioned relative to the viewport. It stays fixed even when the page is scrolled.

1. CSS units

* Absolute units (always the same size): px
* Relative units (relative to sth else, eg the size of the parent font, size of viewport): em, rem, %, vh, vw
  + em dựa vào font-size của parent gần nhất
  + rem dựa vào font-size của html
  + % dựa vào parent gần nhất
  + vh = 1% viewport height; vw = 1% viewport width

Flexbox Crash Course

1. Before we begin

* This course is for beginner
* Continuation of the CSS crash course
* We’re gonna learn the important concepts in flexbox

1. What is Flexbox

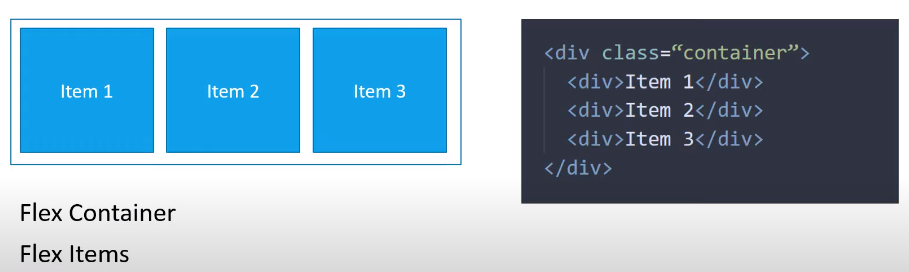
* Is a one-dimensional layout model

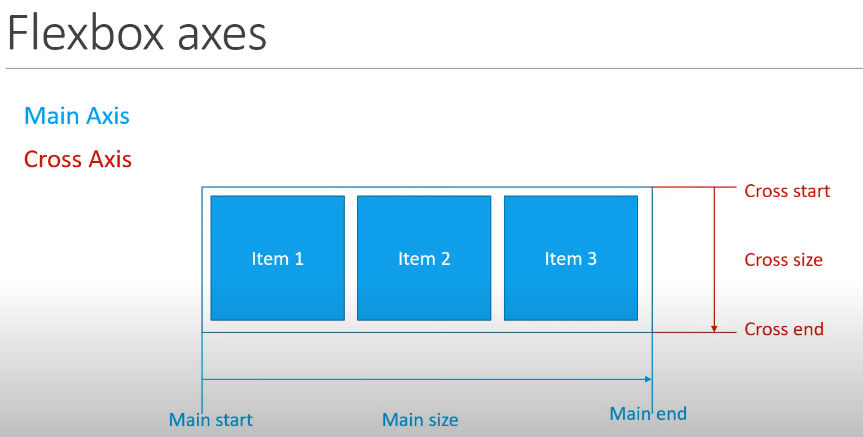
1. Layout modes before Flexbox

* Block & Inline
* Position

1. Terminology\*

* Flex Container & Flex Items





* Understanding Flexbox is basically getting familiar with the props associated with the flex container and the flex items

1. Flex Container Props
   * + 1. display: flex
       2. flex-direction
       3. flex-wrap
       4. flex-flow = flex-direction + flex-wrap
       5. column-gap
       6. row-gap
       7. gap = row-rap + column-gap
       8. justify-content: align along main axis
       9. align-items: align along cross axis
       10. align-content: align along cross axis, but for multiple line of flex items