



HackSeries 01

CSS

What is CSS ?

CSS (Cascading Style Sheets) styles and layouts web pages by defining how HTML elements are displayed. It separates content from design, using rules with selectors, properties, and values to apply styles like colors, fonts, and spacing. Key concepts include the box model, cascading rules, positioning, responsive design with media queries, and layout systems like Flexbox and Grid. CSS enhances web aesthetics, responsiveness, and user experience efficiently.

Key Concepts of CSS :

- **Selectors:**

Define the elements you want to style. Examples:

- o element selector: `p { color: red; }`
- o class selector: `.btn { font-size: 14px; }`
- o id selector: `#header { background: blue; }`

- **Properties and Values:**

In CSS, **properties** define the aspects of an element's appearance, while **values** specify how those aspects should be styled.

Common Properties and Their Values:

- **color:** Defines the text color.
 - o Example: `color: blue;`
- **font-size:** Specifies the font size.
 - o Example: `font-size: 16px;`

CSS Selector	CSS	HTML
Tag name	<code>h1 { color: red; }</code>	<code><h1>Today's Specials</h1></code>
Class attribute	<code>.large { font-size: 16pt; }</code>	<code><p class="large">...</code>
Tag and Class	<code>p.large {...}</code>	<code><p class="large">...</code>
Element id	<code>#p20 { font-weight: bold; }</code>	<code><p id="p20">...</code>

CSS Pseudo Selectors

hover - Apply rule when mouse is over element (e.g. tooltip)

```
p:hover, a:hover {
  background-color: yellow;
}
```

a:link, a:visited - Apply rule when link has been visited or not visited (link)

<pre>a:visited { color: green; }</pre>	<pre>a:link { color: blue; }</pre>
--	--------------------------------------

CSS Properties

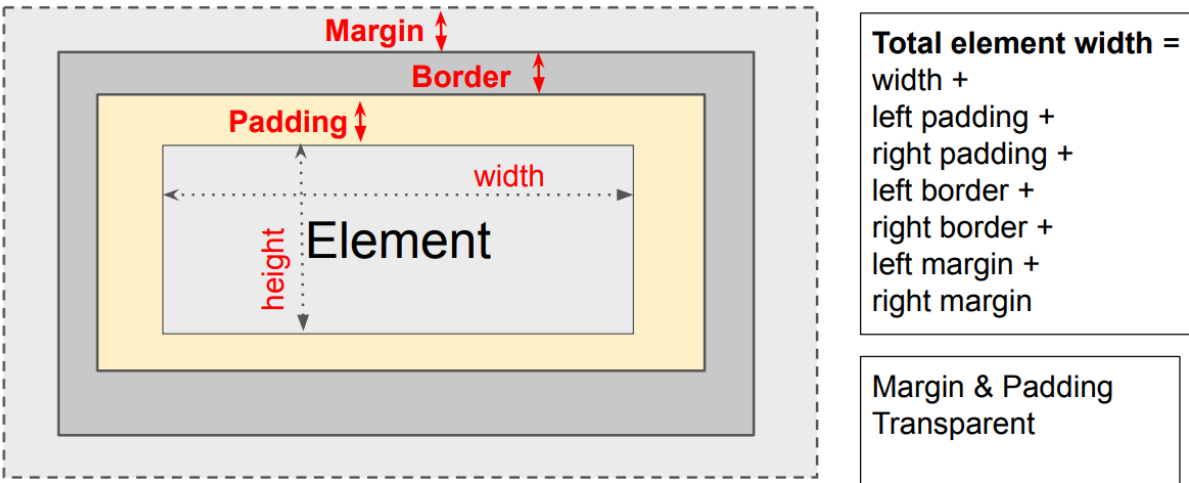
Control many style properties of an element:

- Coloring
- Size
- Position
- Visibility
- Many more: (e.g. `p { text-decoration: line-through; }`)
- Also used in animation

CSS DISTANCE UNITS :

Absolute	
2px	pixels
1mm	millimeters
2cm	centimeters
0.2in	inches
3pt	printer point 1/72 inch
Relative	
2em	2 times the element's current font size
3rem	3 times the root element's current font size

CSS Box Model



Element visibility control properties

`display: none;` - Element is not displayed and takes no space in layout.

`display: inline;` - Element is treated as an inline element.

`display: block;` - Element is treated as a block element.

`display: flex;` - Element is treated as a flex container.

`display: grid;` - Element is treated as a grid container.

`visibility: hidden;` - Element is hidden but space still allocated.

`visibility: visible;` - Element is normally displayed

position property

`position: static;` (default) - Position in document flow

`position: relative;` Position relative to default position via top, right, bottom, and left properties

`position: fixed;` Position to a fixed location on the screen via top, right, bottom, and left properties

`position: absolute;` Position relative to ancestor absolute element via top, right, bottom, and left properties

Fixed position (0,0) is top left corner

Flexbox and Grid layout

- `display: flex;` (Flexbox)
- `display: grid;` (Grid) newer layout method
 - Items flex to fill additional space and shrink to fit into smaller spaces.
 - Useful for web app layout:
 - Divide up the available space equally among a bunch of elements
 - Align of different sizes easily
 - Key to handling different window and display sizes
- Flexbox - Layout one dimension (row or column) of elements
- Grid - Layout in two dimensions (rows and columns) of elements
- Covered in discussion section

Some more common properties

`background-image:` image for element's background

`background-repeat:` should background image be displayed in a repeating pattern (versus once only)

`font, font-family, font-size, font-weight, font-style:` font information for text

`text-align, vertical-align:` Alignment: `center, left, right`

`cursor` - Set the cursor when over element (e.g. `help`)

IMPORTANT LINKS :

1. [UI UX Verse](#)
2. [CSS Tricks](#)
3. [Color Picker](#)
4. [CSS Tutorial](#)

Conclusion:

In conclusion, CSS is an essential tool for designing visually engaging and well-structured web pages. It offers flexibility, allowing developers to control the layout, style, and responsiveness of a website. Through its powerful properties, values, and advanced features like Flexbox, Grid, and media queries, CSS ensures a smooth and dynamic user experience across various devices and screen sizes. Mastering CSS is crucial for creating aesthetically pleasing and functional web interfaces that cater to modern web design standards.