

CSS Fundamentals Session

1. What is CSS

Cascading Style Sheets, fondly referred to as CSS, is a simply designed language intended to simplify the process of making web pages presentable. CSS allows you to apply styles to web pages. More importantly, CSS enables you to do this independent of the HTML that makes up each web page.

CSS is easy to learn and understood, but it provides powerful control over the presentation of an HTML document.

2. Adding CSS to HTML

External CSS is the most common method you will come across, and it involves creating a separate file for the CSS and linking it inside of an HTML's opening and closing <head> tags with a self-closing <link> element:

```
<!-- index.html -->

<head>

    <link rel="stylesheet" href="styles.css" />

</head>

/* styles.css */

div {

    color: white;

    background-color: black;

}

p { color: red; }
```

Internal CSS (or embedded CSS) involves adding the CSS within the HTML file itself instead of creating a completely separate file. With the internal method, you place all of the rules inside of a pair of opening and closing <style> tags, which are then placed inside of the opening and closing <head> tags of your HTML file. Since the styles are being placed directly inside of the <head> tags, we no longer need a <link> element that the external method requires.

```

<head>
  <style>
    div {
      color: white;
      background-color: black;
    }
    p { color: red; }
  </style>
</head>
<body>...</body>

```

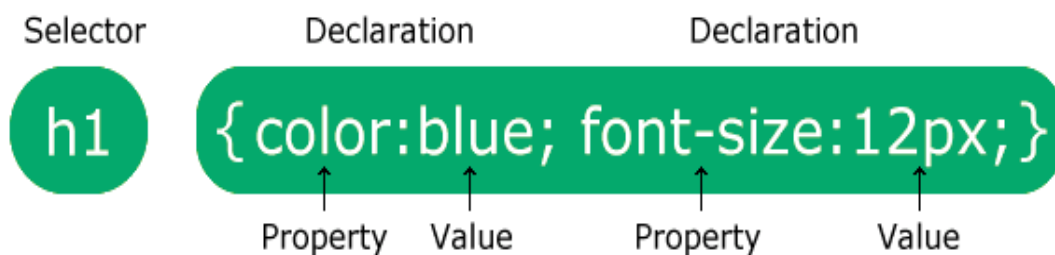
Inline CSS makes it possible to add styles directly to HTML elements, though this method isn't as recommended:

```

<body>
  <div style="color: white; background-color:
black;">
    ...</div>
</body>

```

3. Syntax



4. Selectors

<https://www.theodinproject.com/paths/foundations/courses/foundations/lessons/css-foundations#selectors>

5. Combining selectors

Selector	Description
div span	Descendant selector (all s that are descendants of a <div>)
div > span	Child selector (all s that are a direct child of a <div>)
a ~ span	General Sibling selector (all s that are siblings after an <a>)
a + span	Adjacent Sibling selector (all s that are immediately after an <a>)

<https://www.thoughtco.com/grouping-multiple-css-selectors-3467065>

6. Different properties:

Most commonly used CSS properties:

<https://zellwk.com/blog/9-important-css-properties-you-must-know/>

Explore more tags at:

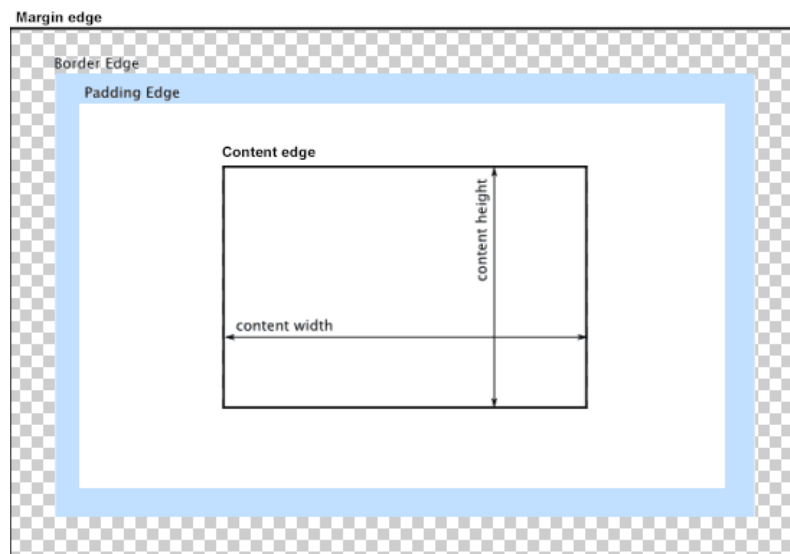
<https://www.tutorialrepublic.com/css-reference/css3-properties.php>

7. Box Model:

Introduction to the CSS basic box model

When laying out a document, the browser's rendering engine represents each element as a rectangular box according to the standard CSS basic box model. CSS determines the size, position, and properties (color, background, border size, etc.) of these boxes.

Every box is composed of four parts (or areas), defined by their respective edges: the content edge, padding edge, border edge, and margin edge.



Content area

The content area, bounded by the content edge, contains the "real" content of the element, such as text, an image, or a video player. Its dimensions are the content width (or content-box width) and the content height (or content-box height). It often has a background color or background image.

If the box-sizing property is set to content-box (default) and if the element is a block element, the content area's size can be explicitly defined with the width, min-width, max-width, height, min-height, and max-height properties.

Padding area

The padding area, bounded by the padding edge, extends the content area to include the element's padding. Its dimensions are the padding-box width and the padding-box height.

The thickness of the padding is determined by the padding-top, padding-right, padding-bottom, padding-left, and shorthand padding properties.

Border area

The border area, bounded by the border edge, extends the padding area to include the element's borders. Its dimensions are the border-box width and the border-box height.

The thickness of the borders are determined by the border-width and shorthand border properties. If the box-sizing property is set to border-box, the border area's size can be explicitly defined with the width, min-width, max-width, height, min-height, and max-height properties. When there is a background (background-color or background-image) set on a box, it extends to the outer edge of the border (i.e. extends underneath the border in z-ordering). This default behavior can be altered with the background-clip css property.

Margin area

The margin area, bounded by the margin edge, extends the border area to include an empty area used to separate the element from its neighbors. Its dimensions are the margin-box width and the margin-box height.

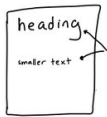
The size of the margin area is determined by the margin-top, margin-right, margin-bottom, margin-left, and shorthand margin properties. When margin collapsing occurs, the margin area is not clearly defined since margins are shared between boxes.

Finally, note that for non-replaced inline elements, the amount of space taken up (the contribution to the height of the line) is determined by the line-height property, even though the borders and padding are still displayed around the content.

8. Units for properties:

JULIA EVANS
@b0rk

CSS units

<p>CSS has 2 kinds of units: absolute & relative</p> <p>absolute: px, pt, pc, in, cm, mm</p> <p>relative: em, rem, ch, ex, vw, vh, %</p>	<p>rem</p> <p>the root element's font size</p> <pre>html { font-size: 14px; }</pre> <p>this means 1rem = 14px everywhere in the document</p>	<p>em</p> <p>the current element's font size</p>  <p>these 2 elements have different values of 1em</p>
<p>0 is the same in all units</p> <pre>.btn { margin: 0; }</pre> <p>you don't need to say 0px or 0em, 0 is always the same</p>	<p>in, cm, mm</p> <p>Great for print stylesheets, not so good for web</p> <p>vw, vh</p> <p>100vw is the viewport width 100vh is the viewport height</p>	<p>rem & em help with accessibility</p> <pre>.modal { width: 20rem; }</pre> <p>this scales nicely if the user increases their browser's default font size</p>

9. Positioning

[Video for static, relative, absolute, fixed and sticky positioning](#)

10. Pseudo Classes

https://www.w3schools.com/css/css_pseudo_classes.asp

<https://css-tricks.com/pseudo-class-selectors/>

11. Check these Projects out

<https://codepen.io/juliangarnier/pen/idhuG>

<https://codepen.io/davidkpiano/pen/GgqLbv>

<https://codepen.io/grizzlyowlet/pen/NGEaxX>

<https://codepen.io/aakashrodrigues/pen/Gfhjw>

12. Practice, Practice, Practice!!

<https://cssbattle.dev/>

<https://www.freecodecamp.org/learn/responsive-web-design/#basic-html-and-html5>

<https://100dayssc.com/about/>

<https://skillcrush.com/blog/html-css-projects/>

13. Need More Resources?

<https://goalkicker.com/CSSBook/CSSNotesForProfessionals.pdf>

<https://www.w3schools.com/css/>

<https://developer.mozilla.org/en-US/docs/Web/CSS>

<https://www.theodinproject.com/paths/foundations/courses/foundations#css-foundations>

