CSS is the language we use to style a Web page.

## **What is CSS?**

* CSS stands for Cascading Style Sheets
* CSS describes how HTML elements are to be displayed on screen, paper, or in other media
* CSS saves a lot of work. It can control the layout of multiple web pages all at once
* External stylesheets are stored in CSS files

## **Why Use CSS?**

CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

## **CSS Solved a Big Problem**

HTML was NEVER intended to contain tags for formatting a web page!

HTML was created to describe the content of a web page, like:

<h1>This is a heading</h1>

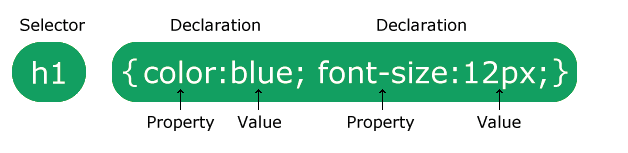
<p>This is a paragraph.</p>

When tags like <font>, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large websites, where fonts and color information were added to every single page, became a long and expensive process.

To solve this problem, the World Wide Web Consortium (W3C) created CSS.

CSS removed the style formatting from the HTML page!

## **CSS Syntax**



**The CSS class Selector**

The class selector selects HTML elements with a specific class attribute.

To select elements with a specific class, write a period (.) character, followed by the class name.

p.center {  
  text-align: center;  
  color: red;  
}

## **The CSS Universal Selector**

The universal selector (\*) selects all HTML elements on the page.

\* {  
  text-align: center;  
  color: blue;  
}

## **The CSS Grouping Selector**

Look at the following CSS code (the h1, h2, and p elements have the same style definitions):

h1, h2, p {  
  text-align: center;  
  color: red;  
}

* Id selectors
* Class selectors
* Element.class selector
* \* - all selectors
* Element selector
* Element, element, …, element

## **Three Ways to Insert CSS**

There are three ways of inserting a style sheet:

* External CSS
* Internal CSS
* Inline CSS

## **External CSS**

<link rel="stylesheet" href="mystyle.css">

## **Inline CSS**

## **Cascading Order**

1. Inline style (inside an HTML element)
2. External and internal style sheets (in the head section)
3. Browser default

## **CSS Color Names**

Color: red, black, …

## **CSS Background Color**

Background-color: …

## **CSS Border Color -> HEX COLOR**

Border: 2px solid red

background-color:rgb(255, 99, 71)

**rgba(*red,* *green*, *blue, alpha*)**

## **Opacity / Transparency**

div {  
  background-color: green;  
  opacity: 0.3;  
}

## **CSS background-image**

background-image: url("paper.gif");

background-repeat: repeat-x;

CSS Borders

* dotted - Defines a dotted border
* dashed - Defines a dashed border
* solid - Defines a solid border
* double - Defines a double border
* groove - Defines a 3D grooved border. The effect depends on the border-color value
* ridge - Defines a 3D ridged border. The effect depends on the border-color value
* inset - Defines a 3D inset border. The effect depends on the border-color value
* outset - Defines a 3D outset border. The effect depends on the border-color value
* none - Defines no border
* hidden - Defines a hidden border

CSS Border Width

border-width property specifies the width of the four borders.

border-width: 25px 10px 4px 35px; /\* 25px top, 10px right, 4px bottom and 35px left \*/

## **CSS Border - Individual Sides**

p {  
  border-top-style: dotted;  
  border-right-style: solid;  
  border-bottom-style: dotted;  
  border-left-style: solid;  
}

CSS Rounded Borders

p {  
  border: 2px solid red;  
  border-radius: 5px;  
}

## **CSS Margins**

CSS has properties for specifying the margin for each side of an element:

* margin-top
* margin-right
* margin-bottom
* margin-left

p {  
  margin-top: 100px;  
  margin-bottom: 100px;  
  margin-right: 150px;  
  margin-left: 80px;  
}

CSS Padding

The CSS padding properties are used to generate space around an element's content, inside of any defined borders.

* padding-top
* padding-right
* padding-bottom
* padding-left

To keep the width at 300px, no matter the amount of padding, you can use the box-sizing property

  box-sizing: border-box;

CSS Height, Width and Max-width

The height and width properties may have the following values:

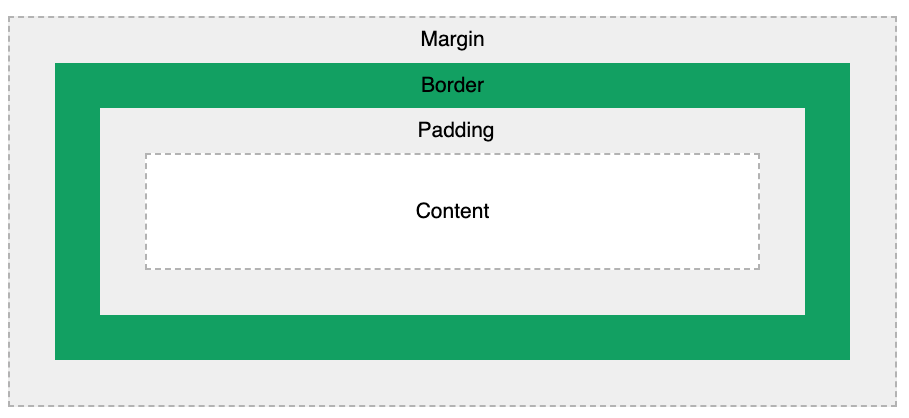
* auto - This is default. The browser calculates the height and width
* length - Defines the height/width in px, cm, etc.
* % - Defines the height/width in percent of the containing block
* initial - Sets the height/width to its default value
* inherit - The height/width will be inherited from its parent value

## **Setting max-width**

The max-width property is used to set the maximum width of an element.

CSS Box Model

In CSS, the term "box model" is used when talking about design and layout



Explanation of the different parts:

* **Content** - The content of the box, where text and images appear
* **Padding** - Clears an area around the content. The padding is transparent
* **Border** - A border that goes around the padding and content
* **Margin** - Clears an area outside the border. The margin is transparent

div {  
  width: 320px;  
  padding: 10px;  
  border: 5px solid gray;  
  margin: 0;  
}

320px (width)  
+ 20px (left + right padding)  
+ 10px (left + right border)  
+ 0px (left + right margin)  
**= 350px**

CSS Text

## **Text Alignment and Text Direction**

In this chapter you will learn about the following properties:

* text-align
* text-align-last
* direction
* unicode-bidi
* vertical-align

## **Text Align Last**

The text-align-last property specifies how to align the last line of a text.

## **Text Direction**

p {  
  direction: rtl;  
  unicode-bidi: bidi-override;  
}

## **Vertical Alignment**

The vertical-align property sets the vertical alignment of an element.

  vertical-align: text-top;

## **Text Transformation**

  text-transform: uppercase;

## **Text Spacing**

  letter-spacing: 5px;

  line-height: 0.8;

  word-spacing: 10px;

## **Text Shadow**

  text-shadow: 2px 2px;

## **The display Property**

Display and visibility

# CSS Icons:

<script src="https://kit.fontawesome.com/a076d05399.js" crossorigin="anonymous"></script>

<i class="fas fa-cloud"></i>  
<i class="fas fa-heart"></i>  
<i class="fas fa-car"></i>  
<i class="fas fa-file"></i>  
<i class="fas fa-bars"></i>

# CSS Links:

# /\* unvisited link \*/ a:link {   color: red; } /\* visited link \*/ a:visited {   color: green; } /\* mouse over link \*/ a:hover {   color: hotpink; } /\* selected link \*/ a:active {   color: blue; }

In addition, links can be styled differently depending on what **state** they are in. The four links states are:

* a:link - a normal, unvisited link
* a:visited - a link the user has visited
* a:hover - a link when the user mouses over it
* a:active - a link the moment it is clicked

style="cursor:auto">

a:link {  
  text-decoration: none;  
}  
  
a:visited {  
  text-decoration: none;  
}  
  
a:hover {  
  text-decoration: underline;  
}  
  
a:active {  
  text-decoration: underline;  
}