

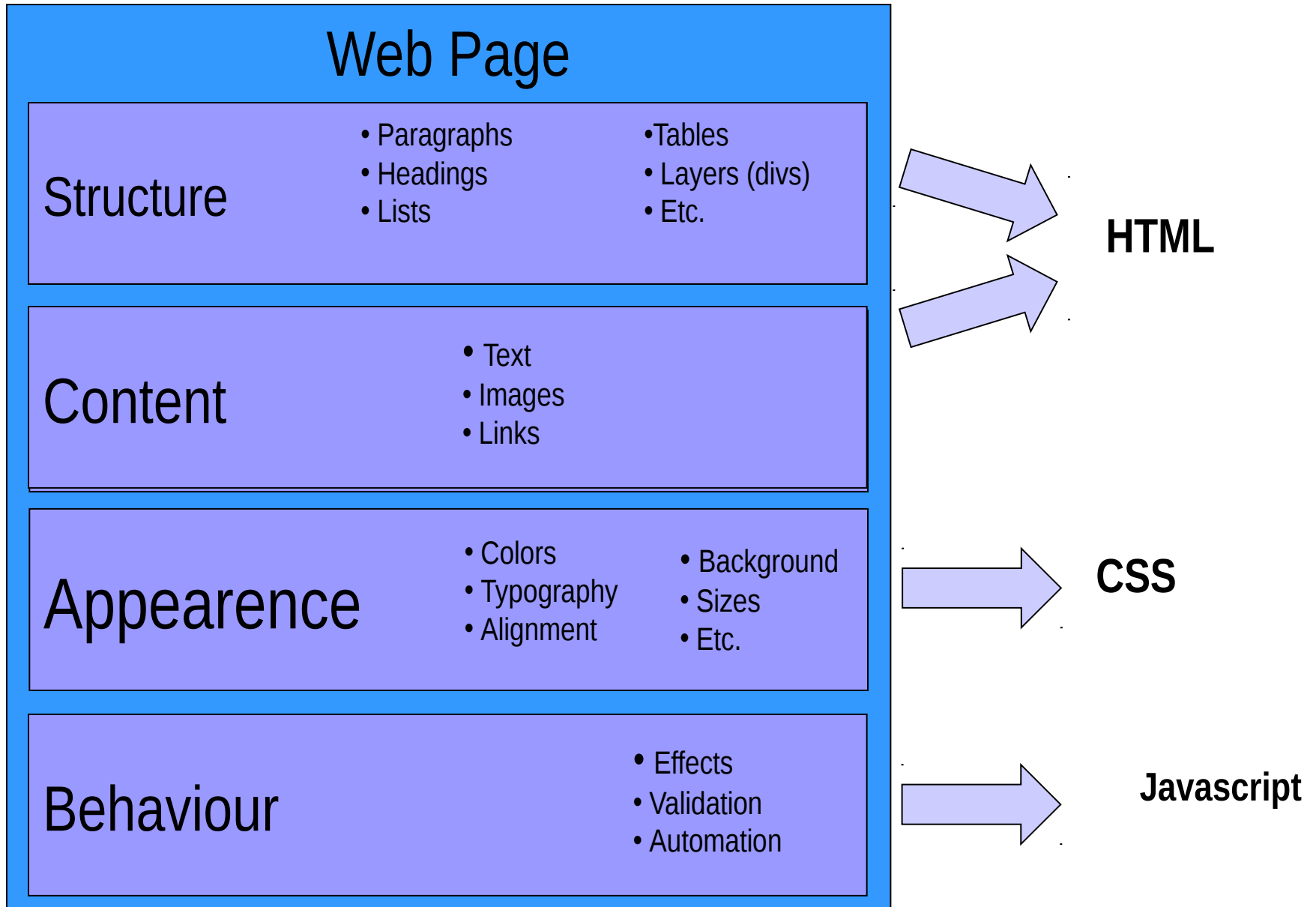


ADVANCED HTML (1)

Cascading Style Sheets

Alex Sánchez

HTML – CSS – Javascript





What is CSS

- CSS complements HTML by providing a look and feel to web pages.
- “raw” HTML pages can look fairly plain, with a default font and font size.
- Using CSS, you can spice up that look,
 - adding color and background images,
 - changing fonts and font sizes,
 - drawing borders around areas, and
 - even changing the layout of the page itself.



The CSS language

- CSS has its own language, separate from HTML, but you wouldn't use CSS without the HTML page.
- In other words, although HTML can stand on its own and present a page to a browser, CSS can't. You wouldn't write a CSS page.
- Rather, you write HTML and then use CSS to help style that page to get it to look like you want it to.



Why use CSS

- Before CSS, an HTML developer changed fonts and colors by changing attributes on each element.
 - If the developer wanted all the headings to look a certain way, he had to change each of those headings.
 - Imagine doing this on a page with ten headings, and then imagine doing it on 50 pages.
- CSS alleviates this burden of individually updating elements and makes it so that *you can apply one single style across one or more elements*.
 - You can apply multiple styles to the same element, and you can target a certain style down to the individual element.
 - For example, if you want all headings to be bold font but a certain heading should have italic, you can do that with CSS.

Separate content from presentation

Without CSS

Formats are set as tag attributes

```
<h2 align="center">
  <font color="blue" size="3"
    face="Times New Roman, serif">
    <i>Web programming</i>
  </font>
</h2>
```

With CSS

Define a new h2 format in a separate style sheet:

```
<style type="text/css">
  h2 {
    text-align: center;
    color: blue;
    font: italic large "Times New Roman", serif;
  }
</style type="text/css">
```

Automatically applied to your header

```
<h2 > Web programming </h2>
```

CSS styles

A CSS is a set of rules.

Each rule is formed by:

- ✓ The Selector (style name)
- ✓ Declaration (defines the style)
 - ✓ Property
 - Value

With CSS we can ...

- ✓ Redefine HTML tag styles
- ✓ Create personalized styles for generic use (Classes)
- ✓ Create styles for a specific HTML tag

```
h2 {
    text-align: center;
    color: blue;
    font: italic large "Times New Roman", serif;
}

.highlightedText{
    font-family: Arial, Helvetica, sans-serif;
    font-size: 12px;
    font-style: normal;
    font-weight: bold;
    color: #000000;
}

#logo {
    background-image: url("/img/logo.gif");
    background-position:center;
    background-repeat:no-repeat;
    height: 50px; width: 150px;
    position: absolute; left: 0px; top: 0px;
```



CSS Syntax



Syntax of CSS

The CSS syntax is made up of 5 parts:

- 1) Selector
- 2) Property/value
- 3) Declaration
- 4) Declaration block
- 5) Curly braces



Selector

Selectors are used to declare which part of the markup a style applies to, a kind of match expression.

3 types of selectors

- 1) Tag selectors (body, p, div, a)
- 2) ID selectors (#wrapper, #sidebar)
- 3) Class selectors (.content, .menu)

*The selector is normally the HTML element you want to style

*Selectors should never start with a number, nor should they have spaces in them

ID Selector

The id selector is used to specify a style for a single, unique element.

The id selector uses the id attribute of the HTML element, and is defined with a "#".

The style rule below will be applied to the element with id="wrapper":

```
#wrapper
{
text-align:center;
border:1px solid red;
Width: 200px;
Height: 100px
}
```

For example, to identify a paragraph as “head”, use the code:

```
<div id="wrapper">... </div>
```

*Do NOT start an ID name with a number!

Class Selector

The class selector is used to specify a style for a group of elements. Unlike the id selector, the class selector is most often used on several elements.

This allows you to set a particular style for many HTML elements with the same class.

The class selector uses the HTML class attribute, and is defined with a "."

In the example below, all HTML elements with class="center" will be center-aligned:

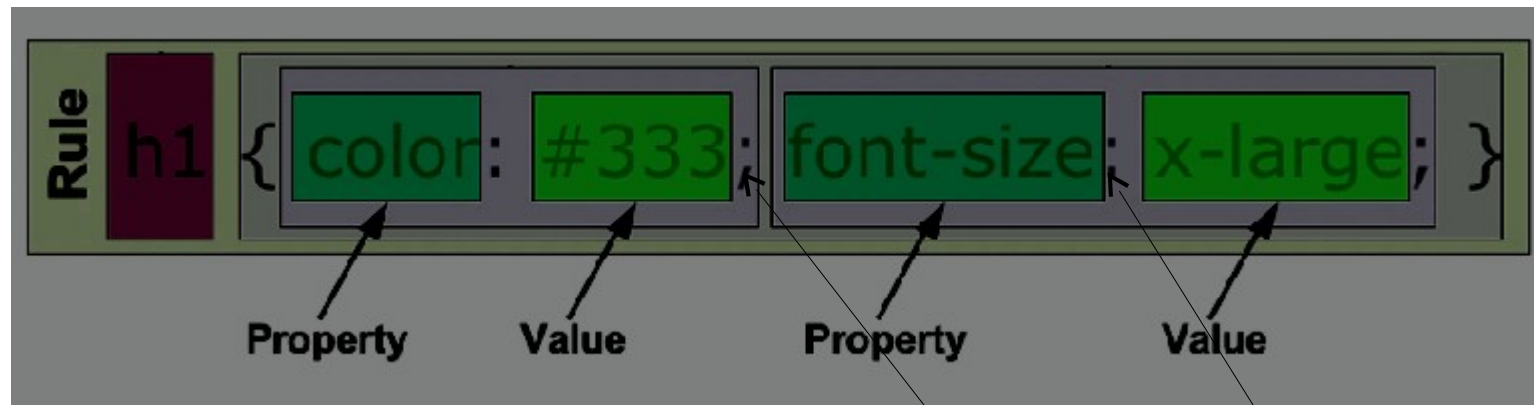
```
.center {text-align:center;} <h1 class="center">Center-aligned heading</h1>  
<p class="center">Center-aligned paragraph.</p>
```

```
p.center {text-align:left;} <p class="left">Center-aligned paragraph.</p>
```

*Do NOT start an ID name with a number!

Property & Value

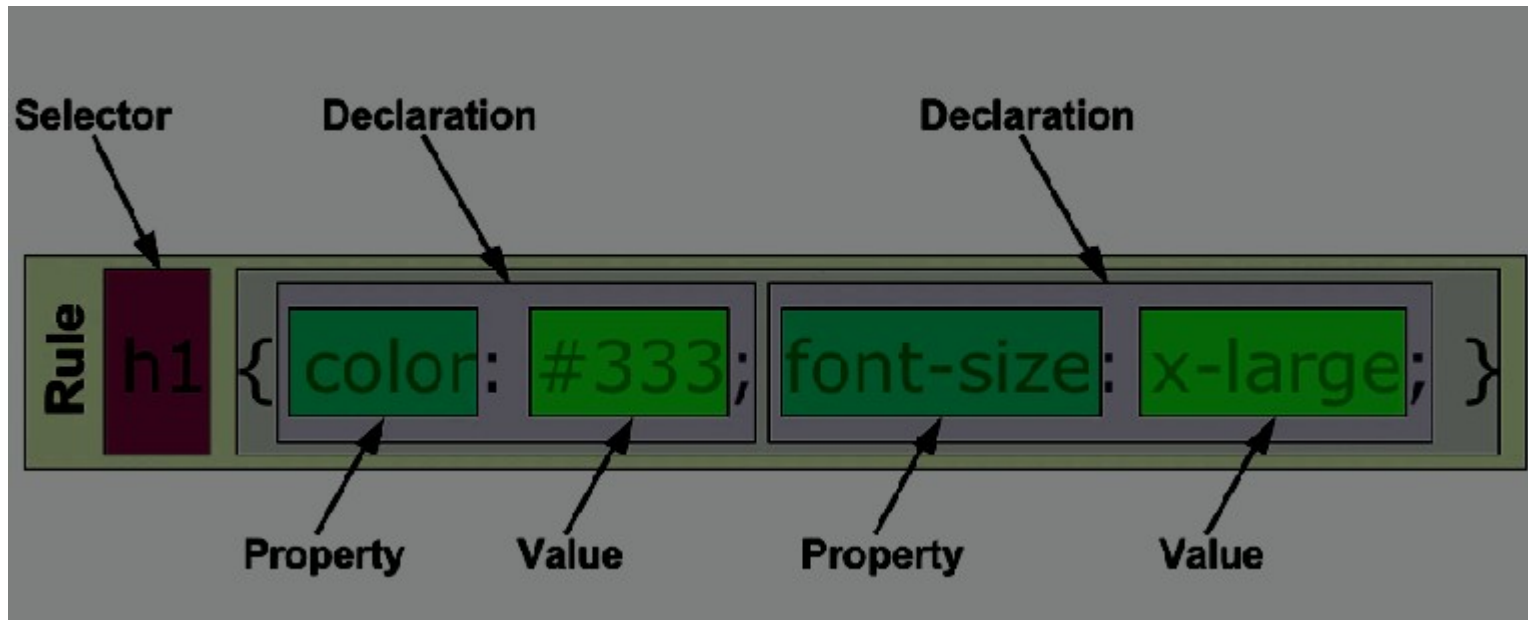
The property is the style attribute you want to change. Each property has a value



- *Properties are separated from their respective values by colons :
- *Pairs are separated from each other by semicolons ;

Declaration

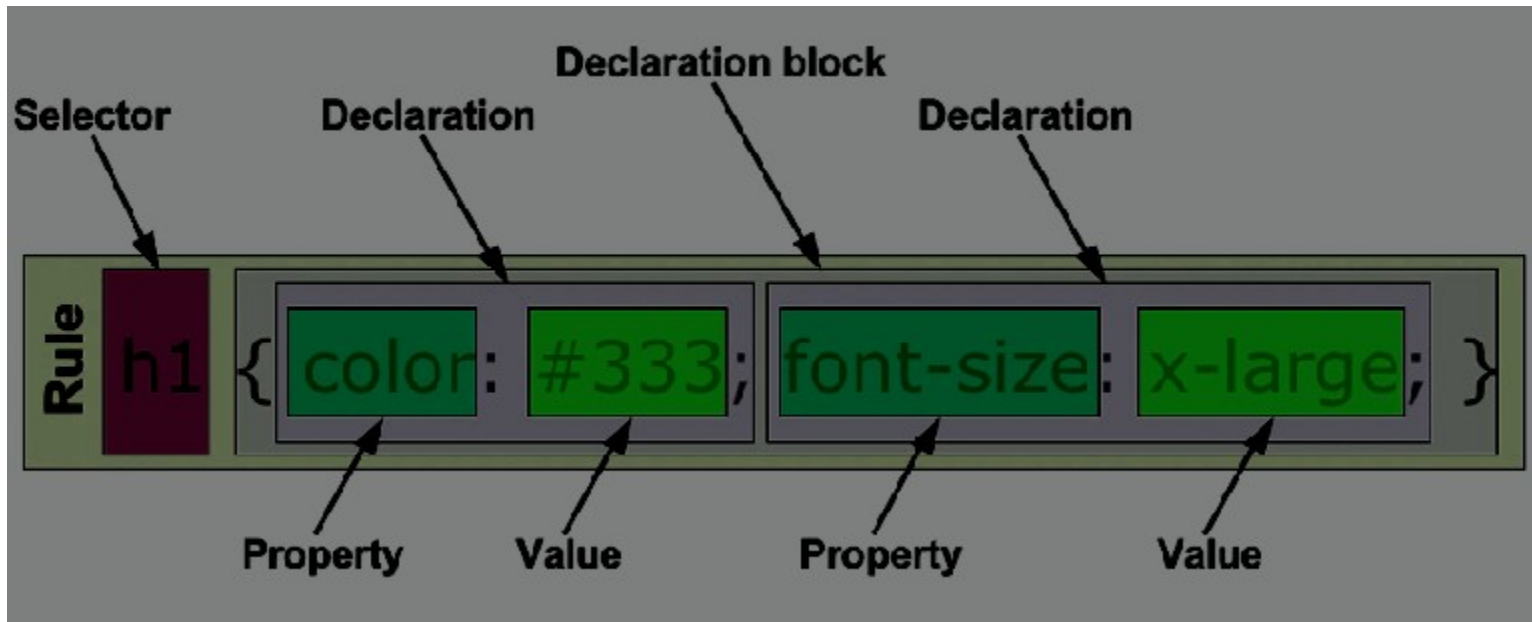
Definition: Each CSS line that includes property and value



*Each declaration consists of a property and a value

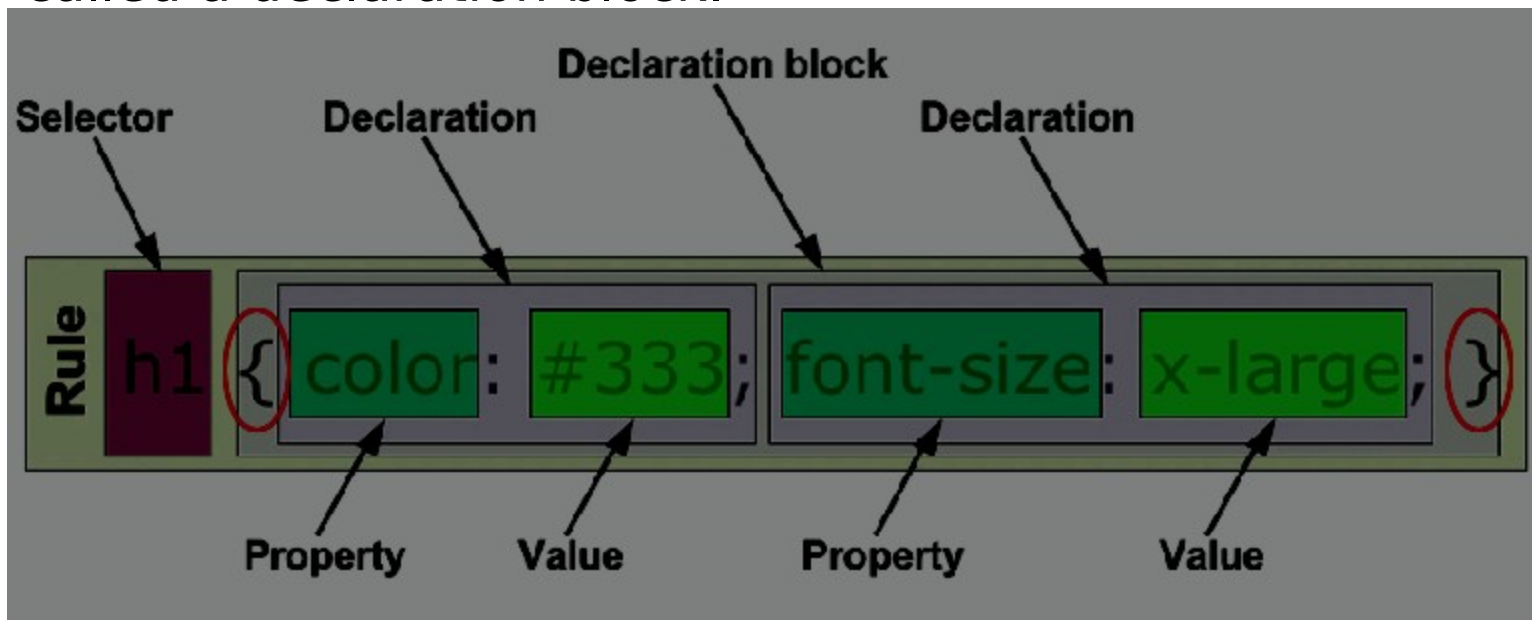
Declaration Block

A declaration block consists of a list of declarations in braces. Each declaration itself consists of a property and a value. If there are multiple declarations in a block.



Curly Braces

The curly braces contain the properties of the element you want to manipulate, and the values that you want to change them to. The curly braces plus their content is called a declaration block.





Applying CSS

How are CSS styles used

✓ Inline Styles

Use the “style” attribute to redefine an HTML element individually.

```
<h2 style="color: blue; background: green;">
  GIE Mod. 2. Web Programming
</h2>
```

✓ Embedded Style

Define CSS rule from within an HTML document.

It can be applied to any element **in this document**.

```
<head>
  <style type="text/css">
    h2 {
      font-style: italic;
      font-weight: bold;
      color: blue;
    }
  </style>
</head>
<body>
  <h2> GIE Mod. 2. Web Programming</h2>
</body>
```

✓ External Style Sheets

Independent CSS files which one references from the HTML documents that are being used

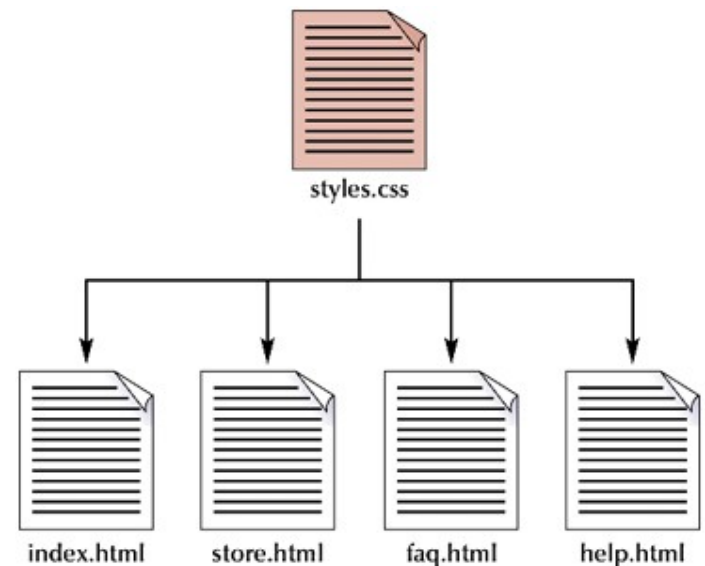
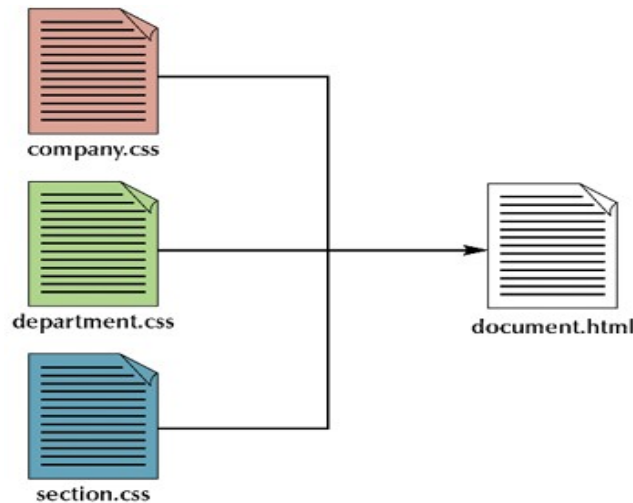
```
<head>
  <link rel="stylesheet" type="text/css" href="estilos.css">
</head>
```

Multiple Style Sheets

You can link a single style sheet to multiple documents in your Web site by using the link element or the **@import** element

You can also link a single document to several style sheets

```
@import "container.css";  
@import "header.css";  
@import "content.css";  
@import "footer.css";
```





Cascading order

What style will be used when there is more than one style specified for an HTML element?

Generally speaking we can say that all the styles will "cascade" into a new "virtual" style sheet by the following rules, where number four has the highest priority:

- 1) Browser default
- 2) External style sheet
- 3) Internal style sheet (in the head section)
- 4) Inline style (inside an HTML element)

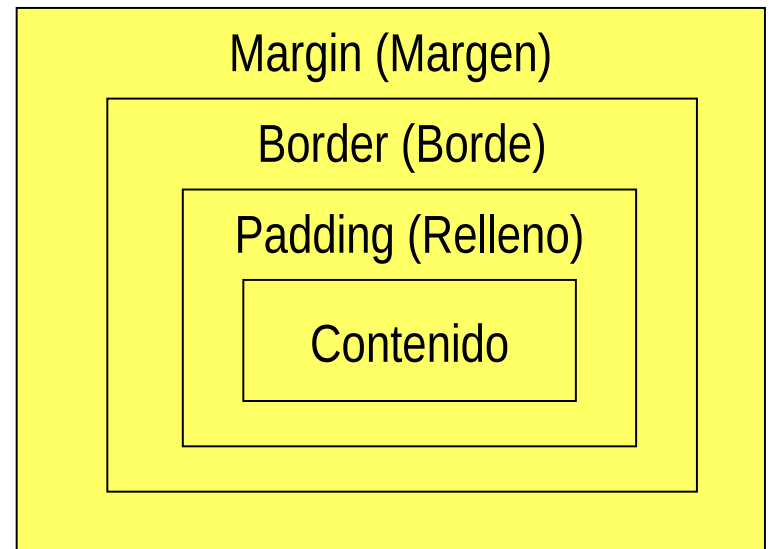
So, an inline style (inside an HTML element) has the highest priority, which means that it will override a style defined inside the <head> tag, or in an external style sheet, or in a browser (a default value).



CSS properties

What can be set CSS

- ✓ Color
- ✓ Font properties
- ✓ Text properties
 - ✓ Word spacing
 - ✓ Alignment
- ✓ Background properties
- ✓ Lists styles
- ✓ Box properties
 - ✓ Margin
 - ✓ Borders
 - ✓ Padding



CSS properties for colors

23

```
p {  
  color: red;  
  background-color: yellow;  
}
```

CSS

This paragraph uses the style above

output

<i>property</i>	<i>description</i>
color	color of the element's text
background-color	color that will appear behind the element

Specifying colors

24

```
p { color: red; }  
h2 { color: rgb(128, 0, 196); }  
h4 { color: #FF8800; }
```

CSS

This paragraph uses the first style above

This h2 uses the second style above.

This h4 uses the third style above.

output

- **color names:** aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white (white), yellow
- **RGB codes:** red, green, and blue values from 0 (none) to 255 (full)
- **hex codes:** RGB values in base-16 from 00 (0, none) to FF (255, full)

Grouping styles

25

```
p, h1, h2 {  
  color: green;  
}  
h2 {  
  background-color: yellow;  
}
```

CSS

This paragraph uses the above style.

This h2 uses the above styles.

output

- A style can select multiple elements separated by commas
- The individual elements can also have their own styles

CSS comments `/*...*/`

26

```
/* This is a comment.  
It can span many lines in the CSS file. */  
p {  
  color: red; background-color: aqua;  
}                                     CSS
```

- CSS (like HTML) is usually not commented as rigorously as programming languages such as Java
- The `//` single-line comment style is NOT supported in CSS
- The `<!-- ... -->` HTML comment style is also NOT supported in CSS

CSS properties for fonts

27

<i>Property</i>	<i>Description</i>
font-family	which font will be used
font-size	how large the letters will be drawn
font-style	used to enable/disable italic style
font-weight	used to enable/disable bold style

Complete list of font properties

(http://www.w3schools.com/css/css_reference.asp#font)

font-family

28

```
p {  
  font-family: Georgia;  
}  
h2 {  
  font-family: "Courier New";  
}
```

CSS

This paragraph uses the first style above.

This h2 uses the second style above.

output

- Enclose multi-word font names in quotes

More about font-family

29

```
p {  
font-family: Garamond, "Times New Roman", serif;  
}
```

CSS

This paragraph uses the above style.

output

- We can specify multiple fonts from highest to lowest priority
- Generic font names:
 - serif, sans-serif, cursive, fantasy, monospace
- If the first font is not found on the user's computer, the next is tried
- Placing a generic font name at the end of your font-family value, ensures that every computer will use a valid font

font-size

30

```
p {  
  font-size: 24pt;  
}
```

CSS

This paragraph uses the style above.

output

- **units: pixels (px) vs. point (pt) vs. m-size (em)**
 - 16px, 16pt, 1.16em
- **vague font sizes:** xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger
- **percentage font sizes, e.g.:** 90%, 120%

font-weight, font-style

31

```
p {  
  font-weight: bold;  
  font-style: italic;  
}
```

CSS

This paragraph uses the style above.

output

- Either of the above can be set to normal to turn them off (e.g. headings)

CSS properties for text

32

property	description
text-align	alignment of text within its element
text-decoration	decorations such as underlining
line-height, word-spacing, letter-spacing	gaps between the various portions of the text
text-indent	indents the first letter of each paragraph

text-align

33

```
blockquote { text-align: justify; }  
h2 { text-align: center; }
```

CSS

The Gollum's Quote

We wants it, we needs it. Must have the precious. They stole it from us.
Sneaky little hobbitses. Wicked, tricky, false!

output

- `text-align` can be `left`, `right`, `center`, or `justify`



text-decoration

34

```
p {  
  text-decoration: underline;  
}
```

CSS

This paragraph uses the style above.

output

- can also be overline, ~~line through~~, blink, or none
- effects can be combined:

```
text-decoration: overline underline;
```

The list-style-type property

35

```
ol { list-style-type: lower-roman; }
```

CSS

□ Possible values:

i. none : No marker

ii. disc (default), circle, square

iii. Decimal: 1, 2, 3, etc.

iv. decimal-leading-zero: 01, 02, 03, etc.

v. lower-roman: i, ii, iii, iv, v, etc.

vi. upper-roman: I, II, III, IV, V, etc.

vii. lower-alpha: a, b, c, d, e, etc.

viii. upper-alpha: A, B, C, D, E, etc.

x. lower-greek: alpha, beta, gamma, etc.

others: hebrew, armenian, georgian, cjk-ideographic, hiragana...

Body styles

36

```
body {  
font-size: 16px;  
}
```

CSS

- Applies a style to the entire body of your page
- Saves you from manually applying a style to each element

CSS properties for backgrounds

37

property	description
background-color	color to fill background
background-image	image to place in background
background-position	placement of bg image within element
background-repeat	whether/how bg image should be repeated
background-attachment	whether bg image scrolls with page
background	shorthand to set all background properties

background-image

38

```
body {  
  background-image: url("images/draft.jpg");  
}
```

CSS

This is the first paragraph

This is the second paragraph...

It occupies 2 lines

- background image/color fills the element's content area

background-repeat

39

```
body {  
background-image: url("images/draft.jpg");  
background-repeat: repeat-x;  
}
```

CSS

This is the first paragraph

This is the second paragraph...

It occupies 2 lines

- can be repeat (default), repeat-x, repeat-y, or no-repeat

background-position

40

```
body {  
background-image: url("images/draft.jpg");  
background-repeat: no-repeat;  
background-position: 370px 20px;  
}
```

CSS

This is the first paragraph

This is the second paragraph...
It occupies 2 lines



- value consists of two tokens, each of which can be top, left, right, bottom, center, a percentage, or a length value in px, pt, etc.
- value can be negative to shift left/up by a given amount



Exercise



Exercise

- Assuming you have created a basic web site with, at least two pages, create a CSS file that modifies, at least:
 - Color, font and text properties
 - Affecting to general elements such as the body and particular ones such as headings
- Link that CSS to all pages in the site
- Create a new CSS file by making some changes to the first one.
 - Link each file to a different page
- Link both CSS to one of the pages. See the effects of cascading