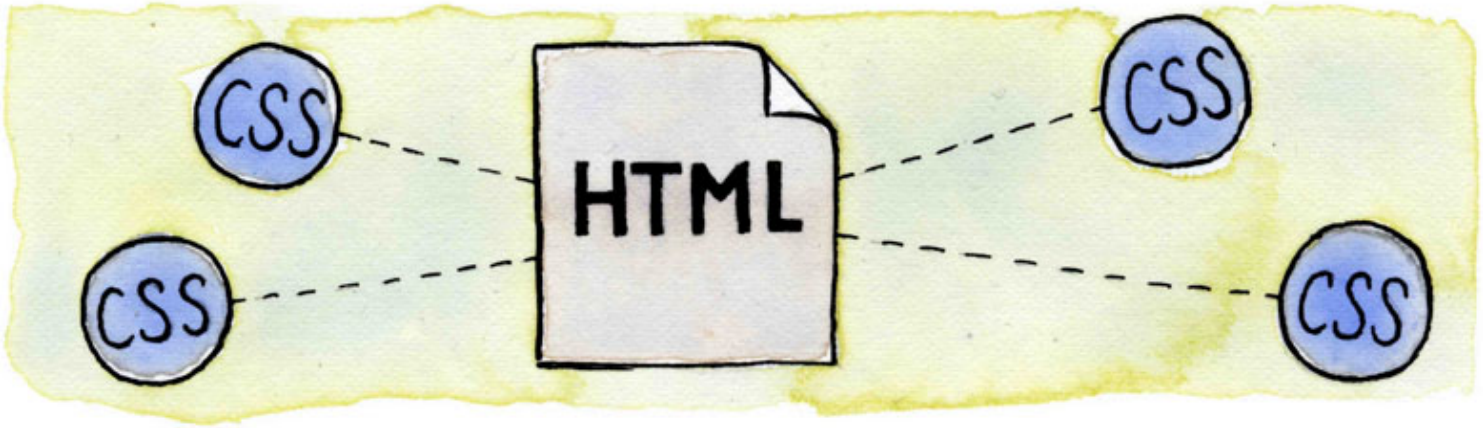
UNIVERSITÀ DI PARMA

CSS



Tecnologie Internet

a.a. 2022/2023

Introduction

HTML was NEVER intended to contain tags for defining the style of a document.

HTML was intended to **display data**, like:
<h1>This is a heading</h1>
This is a paragraph.

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 web sites, where fonts and color information were added to every single page, became a long and expensive process.

Introduction

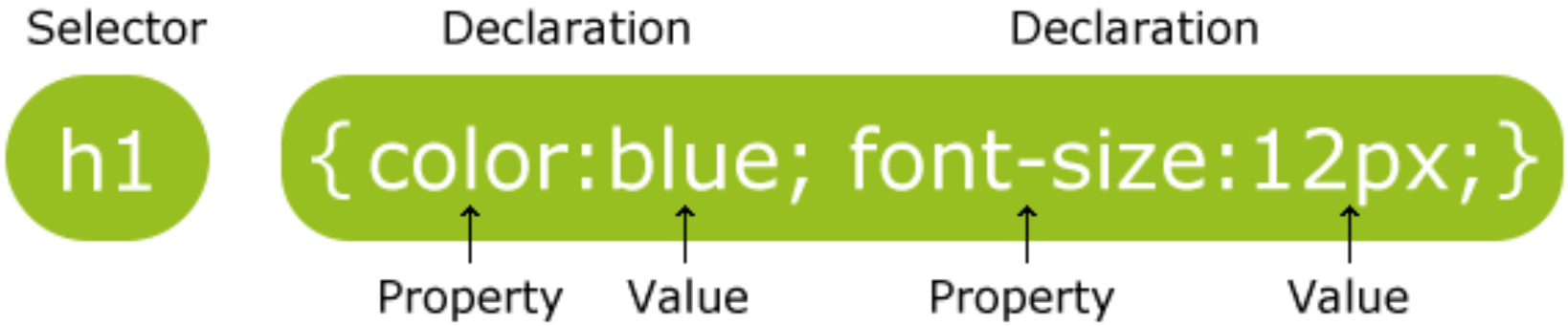
To solve this problem, the World Wide Web Consortium (W3C) created **Cascading Style Sheets (CSS)**.

In HTML 4 and beyond, all formatting could (and should!) be removed from the HTML document, and stored in a separate CSS file.

With an external style sheet file, you can change the look of an entire Web site by changing just one file!

Syntax

A CSS rule set consists of a selector and a declaration block:



The selector points to the HTML element you want to style.
The declaration block contains one or more declarations separated by semicolons.
Each declaration includes a property name and a value, separated by a colon.

CSS Validator: http://jigsaw.w3.org/css-validator/

Selectors

The element selector selects elements based on the element name. You can select all <p> elements on a page like this:

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

The id selector uses the id attribute of an HTML element to select a specific element. The style rule below will be applied to the HTML element with id="para1":

```
#para1 {
    text-align: center;
    color: red;
}
```

How to insert a style sheet

There are three ways of inserting a style sheet:

- **External style sheet**

```
<HEAD>
<LINK rel="stylesheet" type="text/css" href="mystyle.css">
</HEAD>
```

<--- best approach

- **Internal style sheet**

```
<HEAD>
<STYLE>
BODY {
    background-color: linen;
}
H1 {
    color: maroon;
    margin-left: 40px;
}
</STYLE>
</HEAD>
```

- **Inline style**

```
<H1 style="color:blue;margin-left:30px;">This is a heading.</H1>
```

How to insert a style sheet

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
    background-color: linen;
}
h1 {
    color: maroon;
    margin-left: 40px;
}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Background

CSS background properties are used to define the background effects of an element.
CSS properties used for background effects:

- background-color
- background-image
- background-repeat
- background-attachment
- background-position

```
H1 {
    background-color: #6495ed;
}

BODY {
    background-image: url("paper.gif");
}
```

Text

The **color** property is used to set the color of the text.
With CSS, a color is most often specified by:

- a HEX value - like "#ff0000"
- an RGB value - like "rgb(255,0,0)"
- a color name - like "red"

```
BODY {
    color: blue;
}

H1 {
    color: #00ff00;
}

H2 {
    color: rgb(255,0,0);
}
```

Text

The **text-align** property is used to set the horizontal alignment of a text.
Text can be centered, or aligned to the left or right, or justified.

```
H1 {
    text-align: center;
}

P.date {
    text-align: right;
}

P.main {
    text-align: justify;
}
```

Text

The **text-decoration** property is used to set or remove decorations from text.
The text-decoration property is mostly used to remove underlines from links for design purposes:

```
A {
    text-decoration: none;
}
```

It can also be used to decorate text:

```
H1 {
    text-decoration: overline;
}

H2 {
    text-decoration: line-through;
}
```

Text

The **text-transform** property is used to specify uppercase and lowercase letters in a text.
It can be used to turn everything into uppercase or lowercase letters, or capitalize the first letter of each word.

```
P.uppercase {
    text-transform: uppercase;
}

P.lowercase {
    text-transform: lowercase;
}

P.capitalize {
    text-transform: capitalize;
}
```

Text

The **text-indent** property is used to specify the indentation of the first line of a text.

```
P {
    text-indent: 50px;
}
```

Fonts

CSS font properties define the font family, boldness, size, and the style of a text.

- **generic family** - a group of font families with a similar look (like "Serif" or "Monospace")
- **font family** - a specific font family (like "Times New Roman" or "Arial")

The font family of a text is set with the **font-family** property. "Fallback" system: if the browser does not support the first font, it tries the next font. Start with the font you want, and end with a generic family, to let the browser pick a similar font in the generic family, if no other fonts are available.

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

Fonts

Generic family	Font family	Description
Serif	Times New Roman Georgia	Serif fonts have small lines at the ends on some characters
Sans-serif	Arial Verdana	"Sans" means without - these fonts do not have the lines at the ends of characters
Monospace	Courier New Lucida Console	All monospace characters have the same width

Fonts

The **font-style** property is mostly used to specify italic text.
This property has three values:

- normal - The text is shown normally
- italic - The text is shown in italics
- oblique - The text is "leaning" (oblique is very similar to italic, but less supported)

```
P.normal {
    font-style: normal;
}

P.italic {
    font-style: italic;
}

P.oblique {
    font-style: oblique;
}
```

Fonts

The **font-size** property sets the size of the text.
Being able to manage the text size is important in web design.
However, you should not use font size adjustments to make paragraphs look like headings, or headings look like paragraphs.

The font-size value can be an absolute, or relative size.
Absolute size:

- Sets the text to a specified size
- Does not allow a user to change the text size in all browsers (bad for accessibility reasons)
- Absolute size is useful when the physical size of the output is known

Relative size:

- Sets the size relative to surrounding elements
- Allows a user to change the text size in browsers

Fonts

The **font-weight** property specifies the weight of a font:

```
p.normal {
    font-weight: normal;
}

p.thick {
    font-weight: bold;
}

..

<p class="normal">This is a paragraph.</p>
<p class="light">This is a paragraph.</p>
```

Links

Links can be styled with any CSS property (e.g. **color**, **font-family**, **background**, etc.).

```
A {
    color: #FF0000;
}
```

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

```
A:hover {
    color: #FF00FF;
}
```

Lists

The CSS list properties allow you to:
- set different list item markers for ordered lists
- set different list item markers for unordered lists
- set an image as the list item marker

The type of list item marker is specified with the **list-style-type** property:

```
UL.a {
    list-style-type: circle;
}
UL.b {
    list-style-type: upper-roman;
}
```

http://www.w3schools.com/css/tryit.asp?filename=trycss_list-style-type_ex

To specify an image as the list item marker, use the **list-style-image** property.

Tables

To specify table borders in CSS, use the **border** property.

```
TABLE, TH, TD {
   border: 1px solid black;
}
```

http://www.w3schools.com/css/tryit.asp?filename=trycss_table_border

Width and height of a table is defined by the **width** and **height** properties.

```
TABLE {
    width: 100%;
}

TH {
    height: 50px;
}
```

http://www.w3schools.com/css/tryit.asp?filename=trycss_table_width

Tables

The **text-align** property sets the horizontal alignment, like left, right, or center. The **vertical-align** property sets the vertical alignment, like top, bottom, or middle.

```
TH {
    text-align: left;
}

TD {
    height: 50px;
    vertical-align: bottom;
}
```

Text and background colors:

```
TH {
    background-color: green;
    color: white;
}
```

Box model

All HTML elements can be considered as boxes. In CSS, the term "**box model**" is used when talking about design and layout.
The CSS box model is essentially a box that wraps around HTML elements, and it consists of: margins, borders, padding, and the actual content.
The box model allows us to add a border around elements, and to define space between elements.

Box model

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: 300px;
    padding: 25px;
    border: 25px solid navy;
    margin: 25px;
}
```

Box model

Let's style a <DIV> element to have a total width of 400px:

```
DIV {
    background-color: lightgrey;
    width: 320px;
    padding: 30px;
    border: 10px solid navy;
    margin: 0;
}
```

The total width of an element should be calculated like this:
Total element width = width + left padding + right padding + left border + right border + left margin + right margin

The total height of an element should be calculated like this:
Total element height = height + top padding + bottom padding + top border + bottom border + top margin + bottom margin

References

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