

Université de Bordeaux
Licence Parcours International

Initiation au développement Web

HTML 5/CSS 3/Javascript

Marie Beurton–Aimar, Florent Grélard

`florent.grelard@u-bordeaux.fr`

`https://fgrelard.github.io/#teaching`

Version : 5 mars 2023 (diapositives de présentation)



Aim

- Know how to write and to edit Web pages in a formatted language.
- Know how to modify the presentation attributes of a Web page.
- Design couleurstyle sheets

Plan: CSS

1 HTML

2 CSS

- CSS3 Syntax
- Principles
- Selectors
- Text display
- An example

CSS

Définition (CSS)

CSS (*Cascading Style Sheets*) is a language to stylize and format HTML pages.

Standard developed by W3C.

CSS is incremented by **levels** and not by versions

- Current level 3 reuses and extends levels 1 and 2

CSS

Définition (CSS)

CSS (*Cascading Style Sheets*) is a language to stylize and format HTML pages.

Standard developed by W3C.

CSS is incremented by **levels** and not by versions

- Current level 3 reuses and extends levels 1 and 2
- Different from HTML! (HTML versions are different from previous ones)

Principle : separating content from style

Principle

The idea is to separate the content (HTML) from the style (CSS).

CSS stylesheets shape HTML pages. They can be reused for different HTML webpages.

Good practices

A well-designed HTML site should not contain any style information in HTML code.

Plan: CSS

1 HTML

2 CSS

- CSS3 Syntax
- Principles
- Selectors
- Text display
- An example

Simple example

HTML :

```
...  
<h1>Happy Together</h1>  
<p>Wong Kar-Wai movie  
directed in 1997</p>  
...
```

CSS sheet :

```
body {  
    color: #000080;  
}  
  
h1 {  
    font-weight: bold;  
    font-style: italic;  
    color: red;  
    font-size: 24px;  
}
```

Displays :

Simple example

HTML :

```
...  
<h1>Happy Together</h1>  
<p>Wong Kar-Wai movie  
directed in 1997</p>  
...
```

CSS sheet :

```
body {  
    color: #000080;  
}  
  
h1 {  
    font-weight: bold;  
    font-style: italic;  
    color: red;  
    font-size: 24px;  
}
```

Displays :

Happy Together

Wong Kar-Wai movie directed in 1997

General syntax for a CSS stylesheet

Selector

```
h1 {  
    font-weight: bold;  
  
    font-family: Times New Roman, Times, serif;  
}
```

Diagram illustrating the general syntax for a CSS stylesheet:

- Selector**: `h1`
- Property**: `font-weight`
- Value**: `bold`
- Property**: `font-family`
- Values**: `Times New Roman, Times, serif`

Remarques :

- File extension : `.css`
- CSS : no header or body
- Syntax is not sensitive to line breaks or adding spaces
 - ▶ Nevertheless, **indentation is strongly recommended** !

Style inside or outside HTML ? (1/2)

Bad :

```
<h1 style="color:red">Happy Together</h1>
```

Less bad :

```
<head>
...
  <style type="text/CSS" >
    h1 {color:red;}
  </style>
...
</head>
<body>
  <h1>Happy Together</h1>
</body>
```

Style inside or outside HTML ? (2/2)

Good :

```
<head>
...
<link rel="stylesheet" type="text/css"
      media="screen" href="mystyle.css" />
...
</head>

<body>
  <h1>Happy Together</h1>
</body>
```

...where `mystyle.css` is a **separate file** defining properties of `<h1>` (in this case).

Comments

Comments are CSS code which are not executed ; they are **not taken into account by the web browser**

Syntax : `/* Commented text */`

```
body {  
  color: #000080; /* the main text color */  
  /* text-decoration: line-through; */  
}
```

Comments

Comments are CSS code which are not executed ; they are **not taken into account by the web browser**

Syntax : `/* Commented text */`

```
body {  
  color: #000080; /* the main text color */  
  /* text-decoration: line-through; */  
}
```

Warning

Comments are still visible in the **source code** of the CSS file, and by the user.

- Beware not to leave confidential information (passwords, address...)

W3C validator

Just like for HTML, W3C and Mozilla designed a CSS3 code validator.

<http://jigsaw.w3.org/css-validator/>

Plan: CSS

1 HTML

2 CSS

- CSS3 Syntax
- Principles
- **Selectors**
- Text display
- An example

Selectors

A selector is a pattern used to **select the HTML elements** you want to style.

- The targetted HTML elements are the **subject** of the selector.
- Can be based on **various rules** : tag, identifier, class or any combination of those...

Selectors

A selector is a pattern used to **select the HTML elements** you want to style.

- The targetted HTML elements are the **subject** of the selector.
- Can be based on **various rules** : tag, identifier, class or any combination of those...

Tags

- HTML : ` some text `
- CSS : `span { ... }`
- All `` HTML tags are affected

Selectors

A selector is a pattern used to **select the HTML elements** you want to style.

- The targetted HTML elements are the **subject** of the selector.
- Can be based on **various rules** : tag, identifier, class or any combination of those...

Tags

- HTML : ` some text `
- CSS : `span { ... }`
- All `` HTML tags are affected

Attributes

- HTML : ` `
- CSS : `a[href] { ... }`
- Affects all `<a>` tags with `href` attribute.

Identifiers and classes

Identifiers

- HTML : `<h1 id="title">`
- CSS : `h1 #title { ... }`
- **Unique** : only one `id=titre` identifier in the HTML page

Classes

- HTML : `<h1 class="title">`
- CSS : `h1 .title { ... }`
- **Multiple** classes are allowed : different HTML tags can share the same class.

Identifiers and classes : example

HTML :

```
<h2 id="first">As Tears Go By</h2>  
<h2>Wild years</h2>  
<h2 class="blue">Ashes of Time</h2>  
<h2>Chungking Express</h2>  
<h2 class="blue">Fallen Angels</h2>
```

CSS stylesheet :

```
h2 {  
    font-style: italic;  
}  
#first {  
    color: red;  
}  
.blue {  
    color: blue;  
}
```

Identifiers and classes : display

As Tears Go By

Wild years

Ashes of Time

Chungking Express

Fallen Angels

Inheritance

CSS properties are inherited from **parent properties**.

Inheritance by listing properties :

- `h2 #first` inherits from `h2`
- `h2 .blue` inherits from `h2` (but not from `h2 #first`)

Inheritance by parent properties in the HTML document :

- `h2` inherits from `body`

Selector combination

Different syntax can be applied to combine various rules.

Selectors can be tags, classes, identifiers or attributes :

- `selector1, selector2` : select all `selector1` and `selector2` elements.
- `selector1.class1` : select `selector1` which have a class `class1`.
- `selector1 selector2` : select `selector2` which are descendants of `selector1`.
- `selector1 > selector2` : select `selector2` which are direct children of `selector1`.
- `selector1 + selector2` : select `selector2` which are adjacent siblings to `selector1`.
- `selector1 ~ selector2` : select `selector2` which are general siblings to `selector1`.

Selector combination : example

HTML :

```
<h1> Loch Ness Monster </h1>
<h2 class="subsection"> Story
  </h2>
  <div>
    <p class="desc">
      Some text that I want to
      <span id="emph"> highlight
        : </span>
      <a class="desc">link</a>
    </p>
  </div>
  <div>
    <p> Another paragraph </p>
  </div>
```

CSS stylesheet :

```
h2.subsection {
  font-style: italic;
}
.desc {
  color: black;
}
div > .desc {
  color: gray;
}
h2 + div {
  text-align: center;
}
p #emph {
  color: orange;
}
```

Selector combination : display

Displays :

Loch Ness Monster

Story

Some text that I want to highlight: link

Another paragraph

Plan: CSS

1 HTML

2 CSS

- CSS3 Syntax
- Principles
- Selectors
- **Text display**
- An example

Properties : styles

- Property **font-size**
 - ▶ Predefined absolute values : **xx-small**, **x-small**, **medium**, **large**, **x-large**, **xx-large**
 - ▶ Predefined relative values : **smaller**, **larger**
 - ▶ Numerical value : x units, where $\text{unit} \in \{\text{pt}, \text{px}, \text{em}, \%\}$
- Property **font-weight** : controls how thick the font appears
 - ▶ Predefined values : **lighter**, **normal**, **bold**, **bolder**
 - ▶ Numerical values : $x \in [0; 1000]$ (1000 being the “boldest” typography)
- Property **font-style** : **slanting of the font**
 - ▶ Predefined values : **normal**, **italic**, **oblique**
- Property **text-decoration** : **font decoration (underline, overline...)**
 - ▶ Predefined values : **underline**, **overline**, **line-through**, **none**

Properties : font

- Property `font-family` :
 - ▶ Families : serif, sans-serif, `mono`, `monospace`
 - ▶ Font name : `Arial`, `Courier New`, `Courier`, `Helvetica`, `Times New Roman`, `Times`, and many others

| | | |
|-----------------|-----------------|-----------------|
| Quick brown fox | Quick brown fox | Quick brown fox |
| serif | sans-serif | monospace |

Good font practices

- 1 Titles are often displayed in `sans-serif`, and body in `serif`.
- 2 The number of fonts used in a document should be limited (1-2).

Properties : units

- **Absolute** units : cm, mm, in, pt, px
 - ▶ $1\text{in} = 2.54\text{cm} = 25.4\text{mm} = 72\text{pt}$
 - ▶ **px** : relative to the viewing device : $1\text{px} = \text{size of one dot in the viewing device.}$

Properties : units

- **Absolute** units : cm, mm, in, pt, px
 - ▶ 1in = 2.54cm = 25.4mm = 72pt
 - ▶ **px** : relative to the viewing device : 1px = size of one dot in the viewing device.
- **Relative** units : em, vh, vw, %.
 - ▶ **em** : relative to the parent font size : 1em = 100% of the parent font size.
 - ▶ **vw, vh** (viewport width/height) : relative to the width of the viewing device : 1vw : 1% of the viewport width.
 - ▶ **%** : relative to the parent element size

Properties : units

- **Absolute** units : cm, mm, in, pt, px
 - ▶ 1in = 2.54cm = 25.4mm = 72pt
 - ▶ **px** : relative to the viewing device : 1px = size of one dot in the viewing device.
- **Relative** units : em, vh, vw, %.
 - ▶ **em** : relative to the parent font size : 1em = 100% of the parent font size.
 - ▶ **vw, vh** (viewport width/height) : relative to the width of the viewing device : 1vw : 1% of the viewport width.
 - ▶ **%** : relative to the parent element size

Good practices

- In most cases, **relative units** are used.
- For **font sizes**, use **em**.
- For **spacing**, use **%**, **vh** and **vw**.

Properties : colors

- Property `color` : text color

- ▶ Predefined colors : `red`, `blue`, `green`, `orange`, etc.

- ★ Detailed list at : web-color.aliasdmc.fr/

- ▶ User-defined values :

- ★ Hexadécimal : `#RRGGBB`

- ★ RGB : `rgb(r, g, b)` with values ranging from 0 to 255

- ★ RGB : `rgb(r%, g%, b%)` with values ranging from 0 to 100

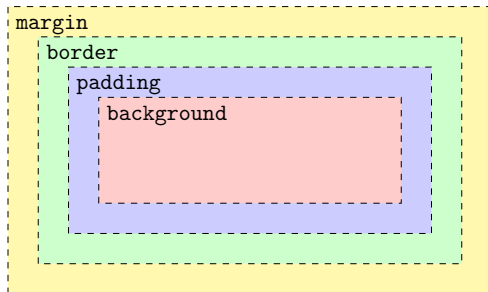
- ★ RGB + alpha : `rgba(r, g, b, a)` with values ranging from 0 to 255, and transparency (`a`) from 0 and 1

- ★ HSL (hue / saturation / light) : `hsl` et `hsla`

Margin and borders

Every HTML element is contained inside a **box**, which is defined by :

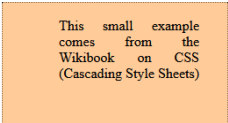
- content area (**background**)
- internal area (**padding**)
- the border (**border**)
- the exterior margin (**margin**)



Margins and borders : example

```
<p class="orangebox">This small example comes from the  
Wikibook on CSS (Cascading Style Sheets)</p>
```

```
p .orangebox {  
  border: 1px dotted black;  
  padding-top: 1em;  
  padding-right: 2em;  
  padding-bottom: 3em;  
  padding-left: 4em;  
  background-color: #fc9;  
  width: 10em;  
  text-align: justify  
}
```



This small example
comes from the
Wikibook on CSS
(Cascading Style Sheets)

Recap : commonly used properties

- `font-style`, `font-family`, `font-weight` : font styling
- `text-align` : align the text inside the parent element (values=left, right, center, justify.)
- `width` : specify the width (in units) of a HTML element
- `color` : text color
- `background-color` : background color
- `padding/margin` : spacing (in units)
- `border` : syntax : *size style color*

Recap : commonly used properties

- `font-style`, `font-family`, `font-weight` : font styling
- `text-align` : align the text inside the parent element (values=left, right, center, justify.)
- `width` : specify the width (in units) of a HTML element
- `color` : text color
- `background-color` : background color
- `padding/margin` : spacing (in units)
- `border` : syntax : *size style color*

Warning

CSS has a lot of capabilities : use the CSS documentation to find out about other properties. <https://www.w3schools.com/cssref>

Plan: CSS

1 HTML

2 CSS

- CSS3 Syntax
- Principles
- Selectors
- Text display
- An example

Without CSS



Marceau Barnard

Student in Bordeaux

I am a recent graduate student fascinated by new technologies. I first acquired a strong background in biology in Paris. Then you can continue to write a brief summary about your background.

26 years old

13, Quai des Salinières

33800 Bordeaux

Contact me

Jobs

2015-? Bus driver

2012-2015 PhD in Biology

Training

2010-2012 International Master @ Sydney

2008-2010 Licence parcours International @ Bordeaux

Same example with CSS



Marceau Barnard

Student in Bordeaux

I am a recent graduate student fascinated by *new technologies*. I first acquired a strong background in biology in Paris. Then you can continue to write a brief summary about your background.

Contact me

26 years old

13, Quai des Salinières

33800 Bordeaux

Jobs

2015-? Bus driver

2012-2015 PhD in Biology

Training

2010-2012 International Master @ Sydney

2008-2010 Licence parcours International @ Bordeaux

References

References

References

- Documentary « Une Contre Histoire des Internets » (J. Goetz et J.M. Manach)

<http://lesinternets.arte.tv/>

- HTML5 specification (28/10/2014)

<http://www.w3.org/TR/html5/>

Sources

- HTML history : Wikipédia > HTML

<http://fr.wikipedia.org/wiki/HTML>

References

References

- **CSS language** (Wikilivres)

https://fr.wikibooks.org/wiki/Le_langage_CSS

- **Cascading Style Sheets** (Wikibooks)

https://en.wikibooks.org/wiki/Cascading_Style_Sheets

Sources

- **CSS specifications** par le W3C

<http://www.w3.org/TR/CSS/>

- **Media Queries**

https://en.wikipedia.org/wiki/Media_queries

Licence

Source des images utilisées (1/2)



Titre : Zygiella web

Auteur : Laura Bassett

Source : https://commons.wikimedia.org/wiki/File:Zygiella_web.jpg

Licence : attribution



Titre : Tim Berners-Lee

Auteur : John S. and James L. Knight Foundation

Source : https://commons.wikimedia.org/wiki/File:Tim_Berners-Lee-Knight-crop.jpg

Licence : CC BY-SA 2.0



Titre : Titre d'onglet (Mozilla Firefox 49.0.1 sous Mac OS)

Auteur : Étienne André

Source : Own work

Licence : CC BY-SA 3.0



Titre : Fenêtre de validation du W3C

Auteur : Étienne André

Source : Own work

Licence : CC BY-SA 3.0



Titre : Formulaire depuis Mozilla Firefox 30

Auteur : Étienne André

Source : Own work

Licence : CC BY-SA 3.0

Source des images utilisées (1/2)



Titre : Zygiella web

Auteur : Laura Bassett

Source : https://commons.wikimedia.org/wiki/File:Zygiella_web.jpg

Licence : attribution



Titre : Affichage HTML+CSS (boîte)

Auteur : Étienne André

Source : Own work

Licence : CC BY-SA 3.0



Titre : Page HTML sans CSS

Auteur : Étienne André

Source : Own work

Licence : CC BY-SA 3.0



Titre : Page HTML avec CSS

Auteur : Étienne André

Source : Own work

Licence : CC BY-SA 3.0

Source des images utilisées (2/2)



Titre : Illustration for responsive web design technique

Auteur : Muhammad Rafizeldi

Source : https://commons.wikimedia.org/wiki/File:Responsive_Web_Design.png

Licence : CC BY-SA 3.0

Source des images utilisées (2/2)



Titre : Lecture audio depuis Mozilla Firefox 24

Auteur : Étienne André

Source : Own work

Licence : CC BY-SA 3.0



Titre : Campus de Villetaneuse (2016)

Auteur : OpenStreetMap / cropped by Étienne André

Source : <http://www.openstreetmap.org/>

Licence : CC BY-SA 2.0

Licence de ce document

Ce support de cours a été modifié selon les termes de la licence Creative Commons **Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0)**



<https://creativecommons.org/licenses/by-nc-sa/3.0/>

Auteur original : Étienne André

