

Philippe GENOUD (LIG-Steamer)

Philippe.Genoud@imag.fr

**M2CCI – M2 GEOMAS 2023-2024**  
**cours PLAI-TW (Technologies du Web)**

# Introduction aux CSS

dernière modification : 06/10/2023 08:50



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/).

# Qu'est-ce que les CSS ?

- Au début du Web mélange du contenu et de la présentation dans code HTML
  - balises définissant le contenu et sa structure `<h1><p><img><ul><li> ...`
  - balises définissent l'apparence `<b><i><font>...`

```
balise <font>
127.0.0.1:5501/index.html
Les anciennes balises HTML <font> <b> et <i>
Lorem ipsum, amet consectetur adipiscing elit. Illo a
harum quibusdam numquam et nulla impedit tenetur corrupti facilis cumque quam suscipit
dicta dignissimos quidem reiciendis rem eligendi amet quas sint perferendis sequi, officia
repellendus! Similique aspernatur neque rerum obcaecati temporibus impedit eos cum
modi.
10 </body>
11 <h2>Les anciennes balises HTML &lt;font&gt; &lt;b&gt; et &lt;i&gt;</h2>
12
13 <p>
14 Lorem ipsum, <font color="FF1CF2" size="+3">amet <b>consectetur</b>
   <i>adipiscing</i> elit</font>. Illo a harum quibusdam numquam et
15 nulla impedit tenetur
16 corrupti facilis cumque quam suscipit dicta dignissimos quidem reiciendis
   rem eligendi amet quas sint perferendis
17 sequi, officia repellendus! Similique aspernatur neque rerum obcaecati
   temporibus impedit eos cum modi.
18 </p>
19 </body>
20 </!DOCTYPE html>
```

Nu Html Checker

Showing results for index.html

Checker Input

Show  source  outline  image report [Options...](#)

Check by [file upload](#) [Parcourir...](#) Aucun fichier sélectionné.

Uploaded files with .html or .xht extensions are parsed using the XML parser.

[Check](#)

Use the Message Filtering button below to display options for hiding/showing particular messages, and to see total counts of errors and warnings.

Message Filtering

- Error** The `<font>` element is obsolete. [Use CSS instead.](#)  
From line 14, column 18: to line 14, column 48  
`em ipsum, <font color="FF1CF2" size="+3">amet <`

Document checking completed.

Source

```
1 <!DOCTYPE html>
```

mdn web docs

References > <font>

**<font>**

**Deprecated:** This feature is no longer recommended. Though some browsers might still support it, it may have already been removed from the relevant web standards, may be in the process of being dropped, or may only be kept for compatibility purposes. Avoid using it, and update existing code if possible; see the [compatibility table](#) at the bottom of this page to guide your decision. Be aware that this feature may cease to work at any time.

The `<font>` [HTML](#) element defines the font size, color and face for its content.

**Warning:** Do not use this element. Use the CSS [Fonts](#) properties to style text.

Attributes

mdn web docs

References > <b>

**<b>: The Bring Attention To element**

The `<b>` [HTML](#) element is used to draw the reader's attention to the element's contents, which are not otherwise granted special importance. This was formerly known as the Boldface element, and most browsers still draw the text in boldface. **However, you should not use `<b>` for styling text or granting importance. If you wish to create boldface text, you should use the CSS `font-weight` property. If you wish to indicate an element is of special importance, you should use the `<strong>` element.**

Try it

# Qu'est-ce que les CSS ?

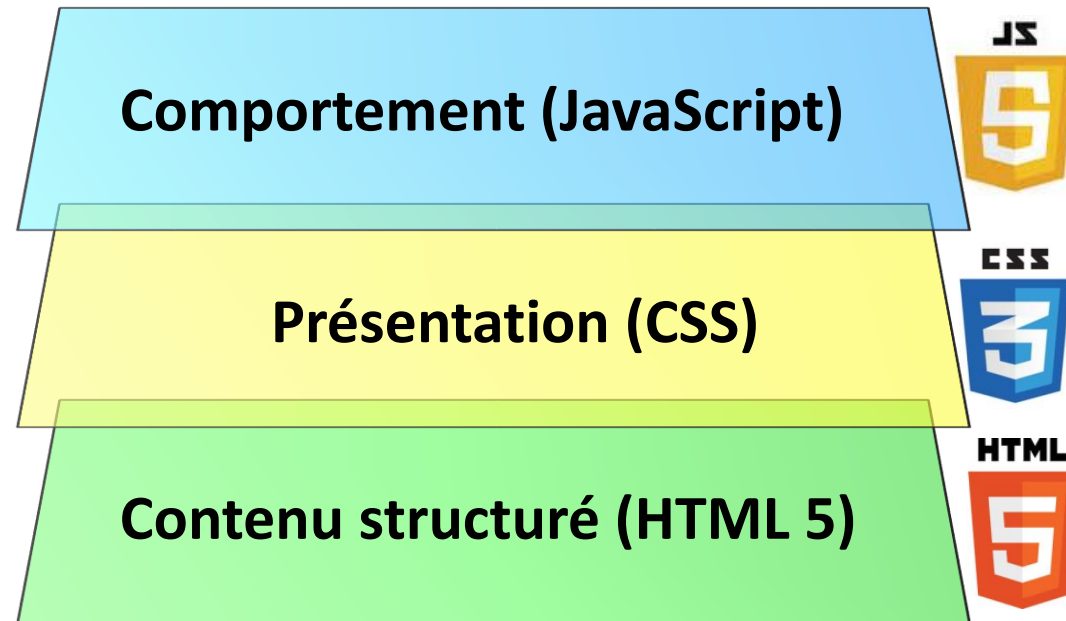


Séparer contenu de la présentation

- HTML (HyperText Markup Language)
  - définir le **contenu** et son **organisation** (paragraphe, tableaux, divisions (sections, articles,...))
- CSS (Cascading Styles Sheets)
  - pour **présenter** les informations
    - définir la couleur, la taille, les bordures, la position ... des éléments de la page HTML,
  - comme HTML **ce n'est pas un langage de programmation**
  - peut être étendu avec Sass/Less
    - préprocesseurs CSS (variables, imbrication des styles, ....)
- Ressources
  - tutorial w3schools : <https://www.w3schools.com/css/default.asp>
  - tutorial Mozilla Developers Network : <https://developer.mozilla.org/fr/docs/Learn/CSS>

# HTML – CSS - JavaScript

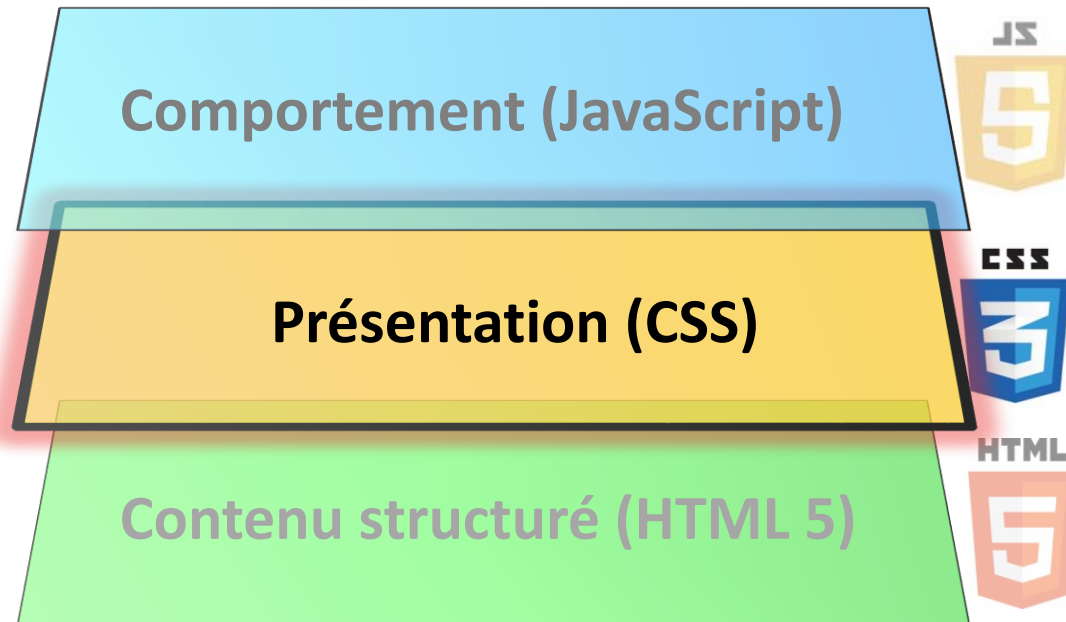
- Décomposer la conception d'un site web en 3 couches



*Head First Mobile Web, Lyza Danger Gardner, Jason Grigsby, O'Reilly 2012*

# CSS

## Cascading Styles Sheets



# Styles CSS

- un ensemble de règles qui indiquent au navigateur
  - comment il doit afficher les différents éléments de la page (*design*)
  - comment ces éléments se positionnent les uns par rapport aux autres (*layout*).



# Exemples d'utilisation des CSS

Séparer contenu de la présentation

<http://www.csszengarden.com/>

Un même document HTML des feuilles de styles CSS différentes

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>CSS Zen Garden: The Beauty of CSS Design</title>
  <link rel="alternate" type="application/rss+xml" title="RSS" href="http://www.csszengarden.com/feed/" />
</head>
<body id="css-zen-garden">
<div class="page-wrapper">
  <section class="intro" id="zen-intro">
    <header role="banner">
      <h1>CSS Zen Garden</h1>
      <h2>The Beauty of <abbr title="Cascading Style Sheet">CSS</abbr></h2>
    </header>
    <div class="summary" id="zen-summary" role="article">
      <p>A demonstration of what can be accomplished through CSS-based design. Select any style sheet from the list to load it into this page.
      <p>Download the example <a href="/examples/index.html" title="This page's sample HTML file" />html file and <a href="/examples/style.css" title="This page's sample CSS file" />css file.
    </div>
    <div class="preamble" id="zen-preamble" role="article">
      <h3>The Road to Enlightenment</h3>
      <p>Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, broken CSS support, and abandoned browsers.
      <p>We must clear the mind of the past. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP, and the major browser creators.
    </div>
  </section>
</div>
</body>
</html>
```

HTML brut sans CSS

## CSS Zen Garden

### The Beauty of CSS Design

A demonstration of what can be accomplished through CSS-based design. Select any style sheet from the list to load it into this page.

Download the example [html file](#) and [css file](#)

### The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, broken CSS support, and abandoned browsers.

We must clear the mind of the past. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP, and the major browser creators.

The CSS Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the time-honored techniques in new and invigorating fashion. Become one with the web.

### So What is This About?

There is a continuing need to show the power of CSS. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The HTML remains the same, the only thing that has changed is the external CSS file. Yes, really.

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it can truly be, once the reins are placed in the hands of those able to create beauty from structure. Designers and coders alike have contributed to the beauty of the web; we can always push it further.

### Participation

Strong visual design has always been our focus. You are modifying this page, so strong CSS skills are necessary too, but the example files are commented well enough that even CSS novices can use them as

# Exemples d'utilisation des CSS

Séparer contenu de la présentation

<http://www.csszengarden.com/>

Un même document HTML des feuilles de styles CSS différentes

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>CSS Zen Garden: The Beauty of CSS Design</title>
  <link rel="stylesheet" media="screen" href="style1.css">
  <link rel="alternate" type="application/rss+xml" title="RSS" href="http://www.csszengarden.com/feed">
</head>
```

index.html

```
<body id="css-zen-garden">
<div class="page-wrapper">
  <section class="intro" id="zen-intro">
    <header role="banner">
      <h1>CSS Zen Garden</h1>
      <h2>The Beauty of <abbr title="Cascading Style Sheet">CSS</abbr></h2>
    </header>
    <div class="summary" id="zen-summary" role="article">
      <p>A demonstration of what can be accomplished through CSS-based design. Select any style sheet from the list to load it into this page. Download the example <a href="/examples/index.html" href="/examples/style.css" title="This page's sample CSS file">here</a> to see how it works.</p>
    </div>
    <div class="preamble" id="zen-preamble" role="article">
      <h3>The Road to Enlightenment</h3>
      <p>Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, broken CSS support, and abandoned browsers. We must clear the mind of the past. Web enlightenment is the way to a better world. We will download it and pace for our share.</p>
    </div>
  </section>
</div>
```

```
/* basic elements */
html {
  margin: 0;
  padding: 0;
}
body {
  font: 75% georgia, sans-serif;
  line-height: 1.88889;
  color: #555753;
  background: #fff url(http://csszengarden.com/001/bg1.jpg) no-repeat;
  margin-top: 0;
  text-align: justify;
}
```

style1.css

```
h1 {
  display: block;
}
h2 {
  display: none;
}
html {
  font-family: sans-serif;
  -ms-text-size-adjust: 100%;
  -webkit-text-size-adjust: 100%;
}
body {
  margin: 0;
}
```

style2.css



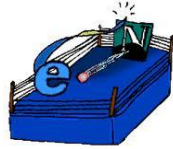


# CSS - Historique

1989 HTML 1.0

1991 HTML 2.0

HTML 3



**Mélange du contenu et de la présentation dans HTML**  
balises <b> <i> <font> ....

1995 HTML 3.2

1996



**CSS1 : 1<sup>ère</sup> recommandation officielle du W3C**

1998 HTML 4.0  
CSS

**CSS2 : extension du concept des feuilles de style**

1999 HTML 4.01



2000 XHTML 1.0

2002

**CSS2.1 : quelques ajustements**

2012 HTML 5



**CSS3 : animations, effets textuels,  
transformations 2D/3D....**

# Objectifs du cours sur les CSS

- CSS sont extrêmement riches et peuvent être complexes
  - 583 propriétés recensées au 04 octobre 2023  
(source <https://www.w3.org/Style/CSS/all-properties.en.html> )
- Il ne s'agit pas de passer en revue toute les propriétés mais de comprendre les principes essentiels des CSS
  - comment ajouter des CSS à une page HTML
  - comment associer des propriétés de styles à des éléments d'une page (sélecteurs)
  - modèle de boites des éléments HTML
  - positionnement des éléments

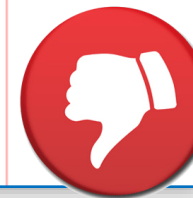
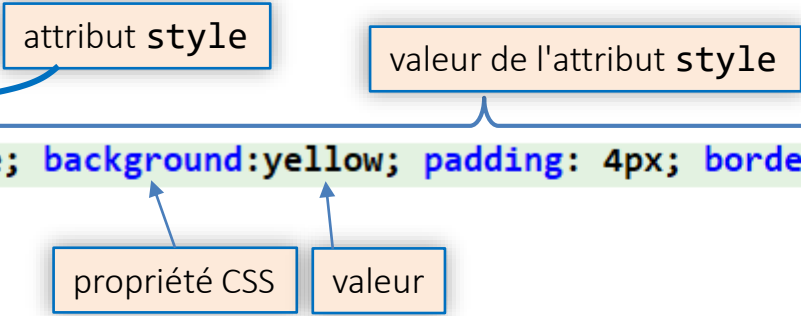
# Comment ajouter des CSS à une page HTML ?

- 3 manières de définir des styles
  - ① inline CSS : directement dans les éléments HTML de la page
  - ② CSS internes : en utilisant des balises `<style>` dans un document HTML
  - ③ CSS externes : en liant au document HTML, une feuille de style (fichier `.css`) externe

# CSS inline

① Style *inline* défini directement dans une balise à l'aide d'un attribut **style**

```
<!DOCTYPE html>
<html>
  <head>
    <title>O1 inline</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
  </head>
  <body>
    <h1>
      Titre sans style
    </h1>
    <h1 style="color:blue; background:yellow; padding: 4px; border:solid 3px green;">
      Titre avec style
    </h1>
    <h1>
      Titre sans style
    </h1>
  </body>
</html>
```



- mélange structure et présentation,
- ne s'applique qu'à une seule balise

ne pas utiliser cette manière de faire

# Styles internes

② Styles définis dans un élément **style** placé dans l'en tête (**head**) du document

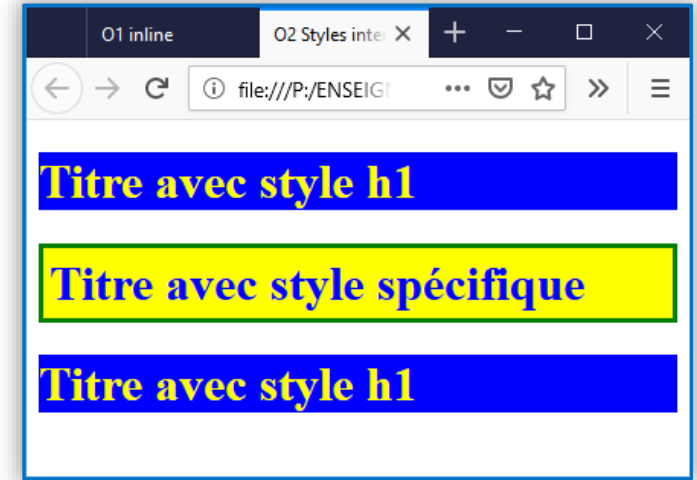
```
<!DOCTYPE html>
<html>
  <head>
    <title>O2 Styles internes</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <style>
      h1 {
        color: yellow;
        background: blue;
      }
      #titreSurligné {
        color: blue;
        background: yellow;
        padding: 4px;
        border:solid 3px green;
      }
    </style>
  </head>
  <body>
    <h1>
      Titre avec style h1
    </h1>
    <h1 id="titreSurligné">
      Titre avec style spécifique
    </h1>
    <h1>
      Titre avec style h1
    </h1>
  </body>
</html>
```

règle de style

propriété

valeur

sélecteur : indique les éléments concernés par la règle



Meilleure séparation du contenu et de la présentation : styles du document localisés en un seul endroit



Selon le sélecteur utilisé un même style peut être appliqué à un ou plusieurs éléments du document



Styles utilisables uniquement à l'intérieur du document où ils sont définis

# Styles externes

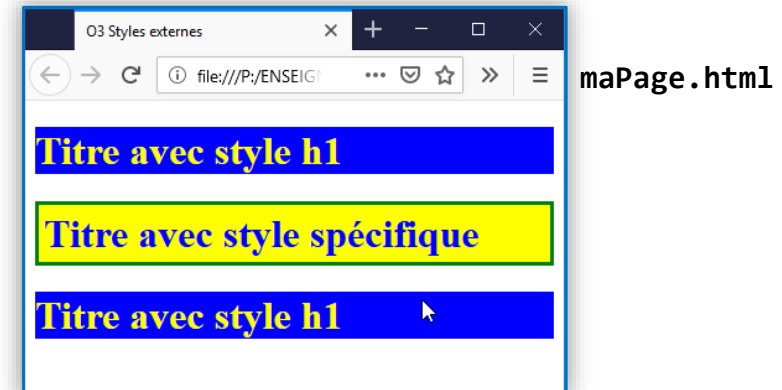
③ Styles définis dans un élément **style** placé dans l'en tête (**head**) du document

```
maPage.html
<!DOCTYPE html>
<html>
  <head>
    <title>O3 Styles externes</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="css/mesStyles.css" rel="stylesheet" type="text/css"/>
  </head>
  <body>
    <h1>
      Titre avec style h1
    </h1>
    <h1 id="titreSurligné">
      Titre avec style spécifique
    </h1>
    <h1>
      Titre avec style h1
    </h1>
  </body>
</html>
```

Lien vers une feuille de style externe (fichier .css)

```
meStyles.css
h1 {
  color: yellow;
  background: blue;
}
#titreSurligné {
  color: blue;
  background: yellow;
  padding: 4px;
  border:solid 3px green;
}
```

```
monAutrePage.html
<!DOCTYPE html>
<html>
  <head>
    <title>O3 Styles externes</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="css/mesStyles.css" rel="stylesheet" type="text/css"/>
  </head>
  <body>
    <h1>
      Titre avec style h1
    </h1>
  </body>
</html>
```



- 👍 Séparation du contenu (fichier .html) et de la présentation (fichier .css)
- 👍 un même style peut être appliqué à un ou plusieurs éléments du document
- 👍 Possibilité de partager les styles entre plusieurs documents

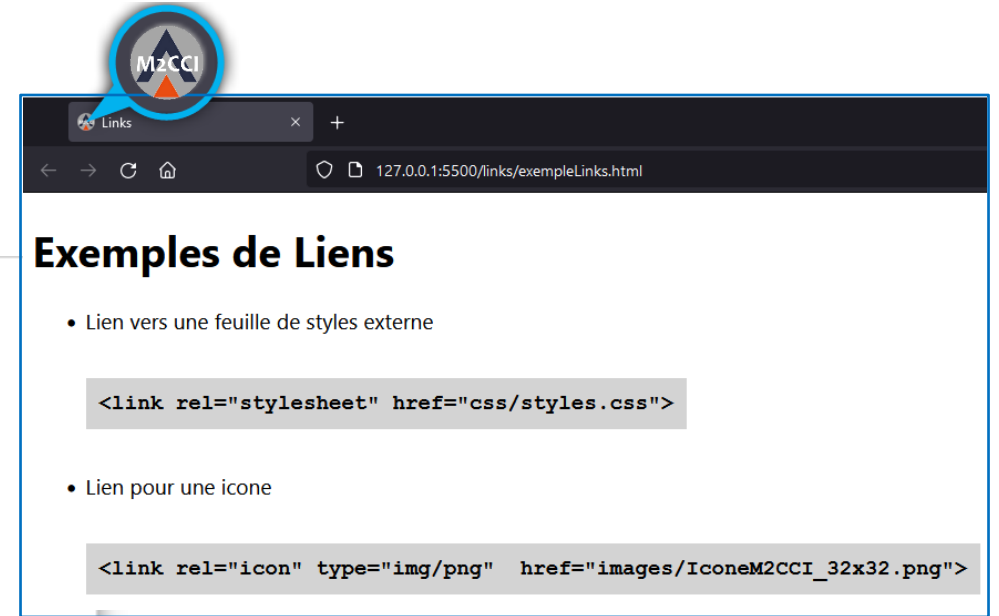
# Styles externes

- La balise `<link>`
  - attribut `rel` : relation entre le document et le document lié (`stylesheet`, `icon`, `licence`...)
  - attribut `href` : la localisation du document lié
  - attribut `type` : le type de média du document lié
  - ... (voir <https://developer.mozilla.org/fr/docs/Web/HTML/Element/link>)

```
exempleLinks.html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Links</title>
  <link rel="stylesheet" href="./css/styles.css">
  <link rel="icon" type="image/png" href="images/IconeM2CCI_32x32.png">
</head>
<body>
  <h1>
    Exemples de Liens
  </h1>
</body>
</html>
```

type implicite (text/css)

type image au format png



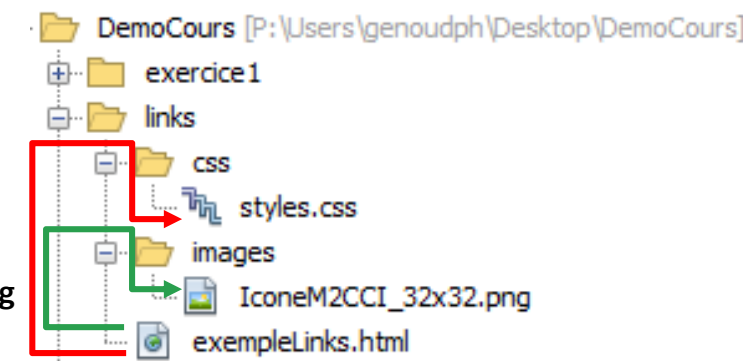
Exemples de Liens

- Lien vers une feuille de styles externe

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

- Lien pour une icone

```
<link rel="icon" type="image/png" href="images/IconeM2CCI_32x32.png">
```



Comme pour les liens hypertexte on utilise **TOUJOURS** un chemin **relatif** \* si les ressources sont sur le même site que le document

\* chemin à partir du répertoire où se trouve le document HTML  
un chemin relatif ne commence **jamais** par un /  
le point ( '.' ) est facultatif

`./css/styles.css`

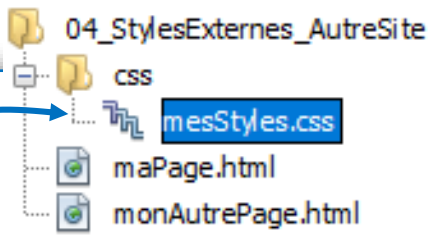
`images/IconeM2CCI_32x32.png`



# Styles externes

maPage.html

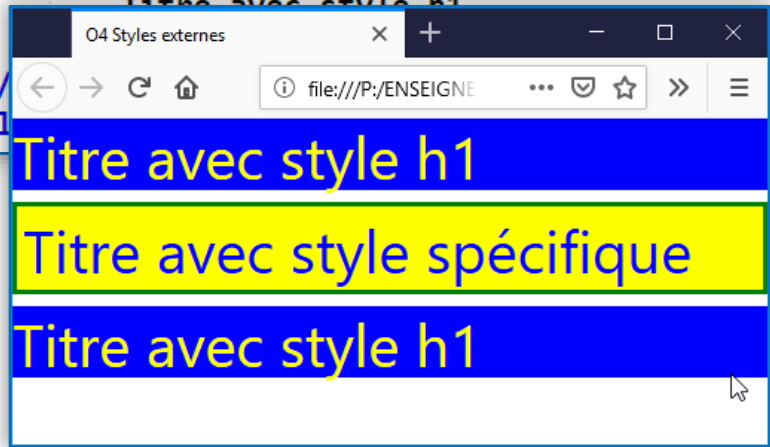
```
<!DOCTYPE html>
<html>
  <head>
    <title>04 Styles externes</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="css/mesStyles.css" rel="stylesheet" type="text/css"/>
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
      integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">
  </head>
  <body>
    <h1>
      Titre avec style h1
    </h1>
    <h1 id="titreSurligné">
      Titre avec style spécifique
    </h1>
    <h1>
      Titre avec style h1
    </h1>
  </body>
</html>
```



• Une page peut avoir plusieurs feuilles de style associées avec possibilité d'avoir des feuilles externes au site

**integrity** (Subresource Integrity ou SRI) permet aux navigateurs de vérifier que les fichiers téléchargés n'ont pas subi de manipulation inattendue.

**crossorigin** indique si la ressource doit être récupérée avec une requête [CORS](#) (Cross-Origin Resource Sharing ou partage des ressources entre origines multiples).

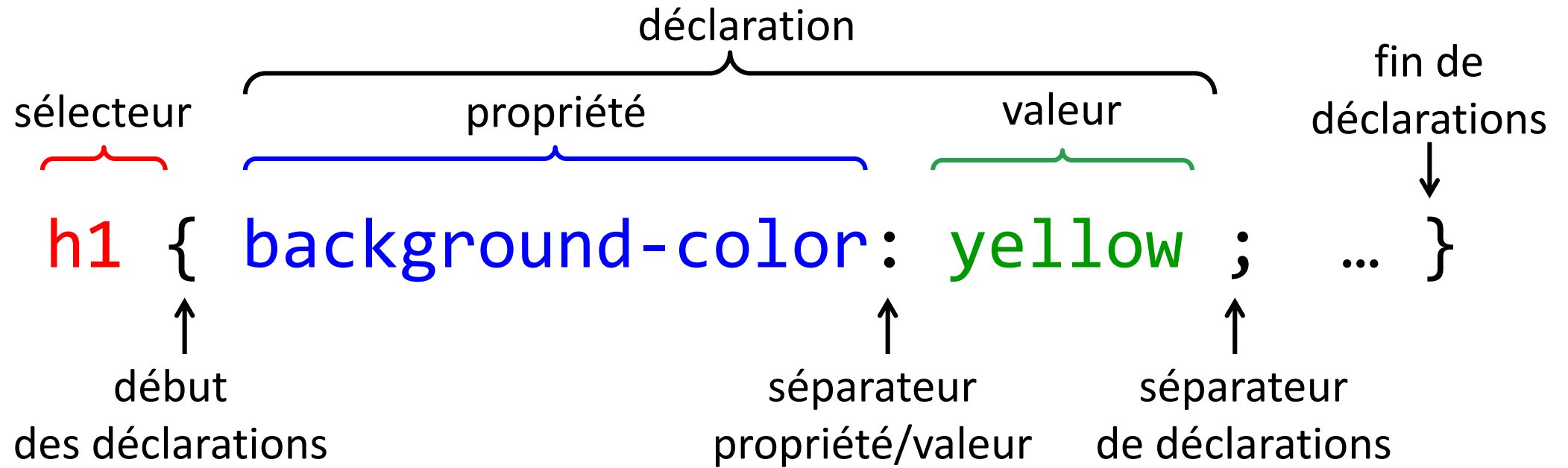


Les styles issus des différentes feuilles de style se combinent



# Définition de styles

- Syntaxe générale d'une règle de style



- **sélecteur** : identifie les éléments HTML de la page web auxquels la règle de style est appliqué
- **propriété** : désigne au navigateur ce qu'il faut modifier
- **valeur** : désigne au navigateur le changement à effectuer

# (Formalisation de la syntaxe

- Notion de langage
  - Langage humain composée de deux choses
    - vocabulaire (ensemble des mots autorisés)
    - grammaire (définit les règles d'assemblage des mots pour construire des phrases)
  - Langages informatiques analogues aux langages naturels
    - décrits par une grammaire rigide
    - composés d'un vocabulaire restreint
- Une grammaire est définie par
  - un ensemble de symboles terminaux (ex : homme, femme, cheval, voiture ...)
  - un ensemble de concepts (ex : phrase, verbe , complément)
  - un concept initial , le concept que l'on doit développer au départ (ex: phrase)
  - un ensemble de règles de grammaire (ex: phrase → sujet verbe complément)

# ... Formalisation de la syntaxe ...

- Diagrammes de Conway

- un type de représentation parmi les plus utilisées pour les langages de programmation de par sa lisibilité et sa souplesse d'expression

- éléments d'un diagramme

- formes arrondies : symboles terminaux



- forme rectangulaire : concepts



- les éléments sont reliés par des traits et flèches

- règles

- chaque diagramme décrit un concept

- un diagramme se lit dans le sens de la flèche

- il existe des formes alternatives, itératives, récursives

- processus de dérivation

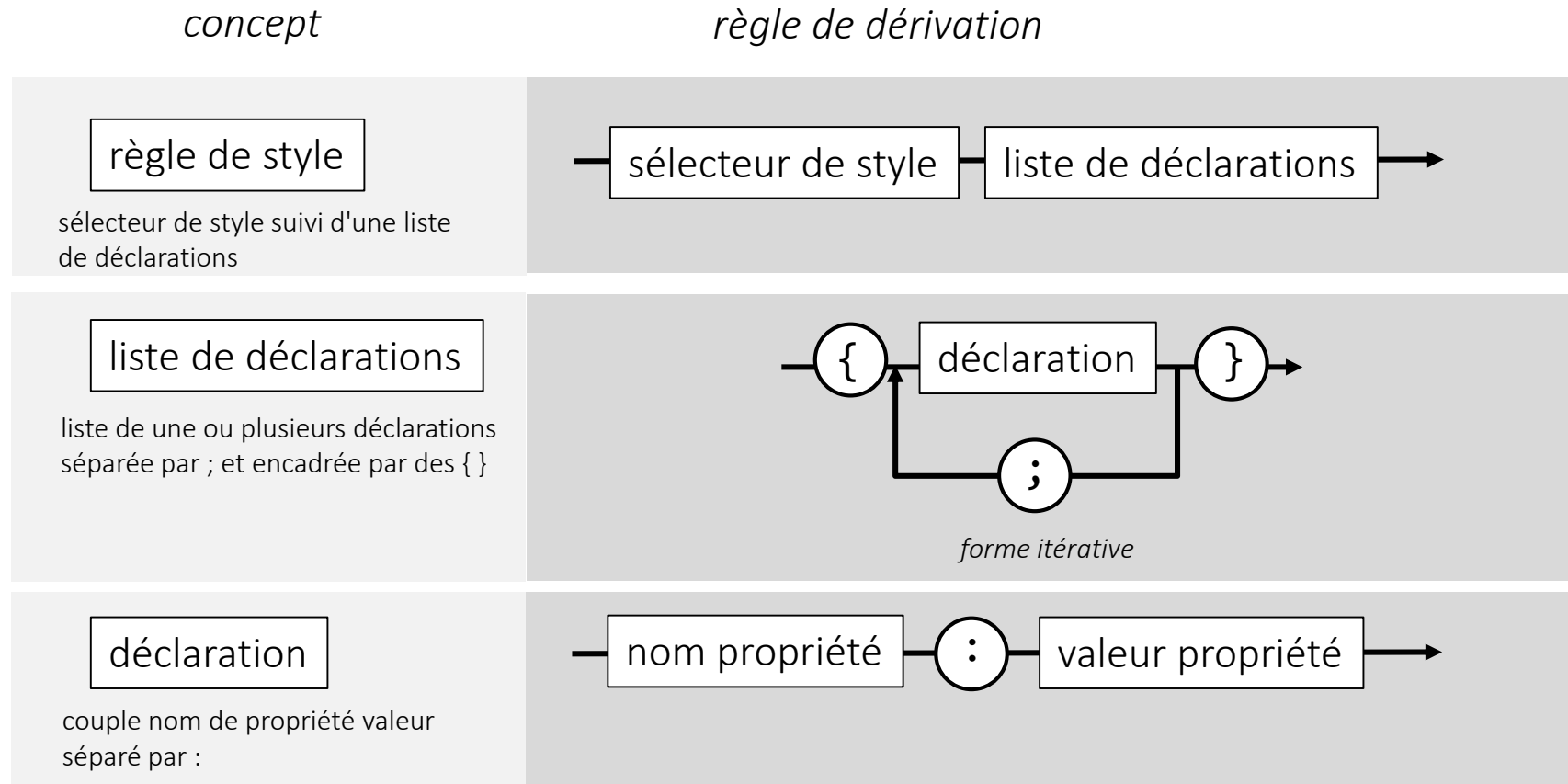
- consiste à remplacer un concept par les éléments rencontrés dans le diagramme qui le décrit

- en partant d'un concept initial on applique le processus de dérivation jusqu'à ce qu'il n'y ait plus que des éléments terminaux dans la phrase

→ on obtient alors une phrase du langage

# ... Formalisation de la syntaxe ...

- Syntaxe générale d'une règle de style sous forme de diagrammes de Conway



# Formalisation de la syntaxe)

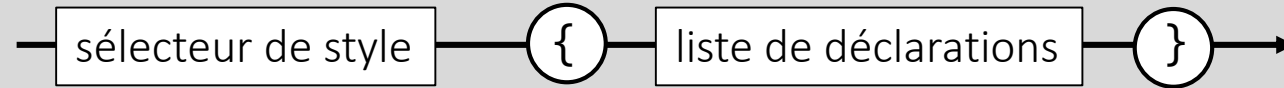
- Syntaxe générale d'une règle de style sous forme de diagrammes de Conway

*concept*

*règle de dérivation*

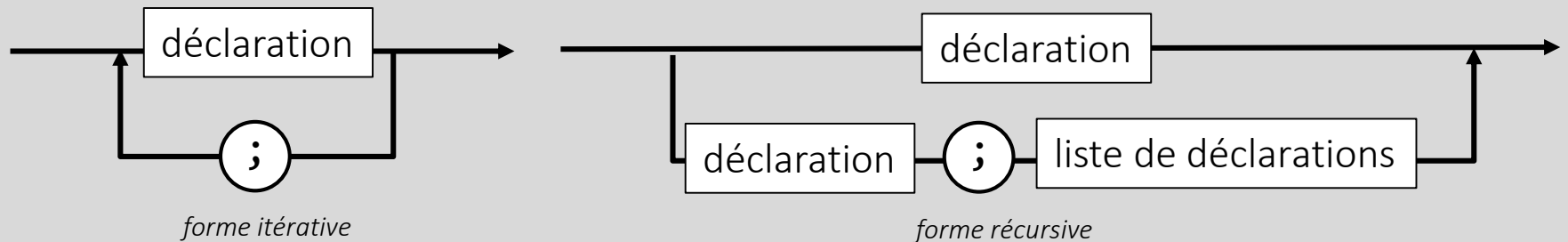
règle de style

sélecteur de style suivi d'une liste de déclarations



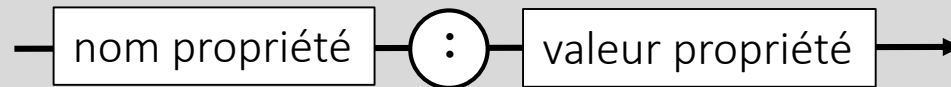
liste de déclarations

liste de une ou plusieurs déclarations séparée par ; et encadrée par des { }



déclaration

couple nom de propriété valeur séparé par :



# Les différents types de sélecteurs

*sélecteur* { *propriété: valeur ; ...* }

- Les sélecteurs sont utilisés pour sélectionner (trouver) les éléments HTML que l'on veut styler
- Les sélecteurs peuvent être regroupés selon 5 catégories
  - Sélecteurs simples
    - sélectionnent des éléments selon leur type, leur id ou leur classe
  - Sélecteurs combinés
    - sélectionnent des éléments selon leur situation dans l'arbre défini par l'imbrication des éléments
  - Sélecteurs de pseudo-classe
    - sélectionnent des éléments en fonction de l'état dans lequel ils se trouvent
  - Sélecteurs de pseudo-élément
    - sélectionnent et stylent une partie d'un élément
  - Sélecteurs d'attributs
    - sélectionnent un élément selon la présence ou la valeur d'un attribut

# Les différents types de sélecteurs

*sélecteur* { *propriété: valeur ; ...* }

- Les sélecteurs sont utilisés pour sélectionner (trouver) les éléments HTML que l'on veut styler
- Les sélecteurs peuvent être regroupés selon 5 catégories
  - **Sélecteurs simples**
    - sélectionnent des éléments selon leur type, leur id ou leur classe
  - Sélecteurs combinés
    - sélectionnent des éléments selon leur situation dans l'arbre défini par l'imbrication des éléments
  - Sélecteurs de pseudo-classe
    - sélectionnent des éléments en fonction de l'état dans lequel ils se trouvent
  - Sélecteurs de pseudo-élément
    - sélectionnent et stylent une partie d'un élément
  - Sélecteurs d'attributs
    - sélectionnent un élément selon la présence ou la valeur d'un attribut

# Sélecteurs simples

Sélecteur de balise (ou d'éléments) style s'appliquant à tous les éléments correspondant à une balise HTML donnée

forme générale

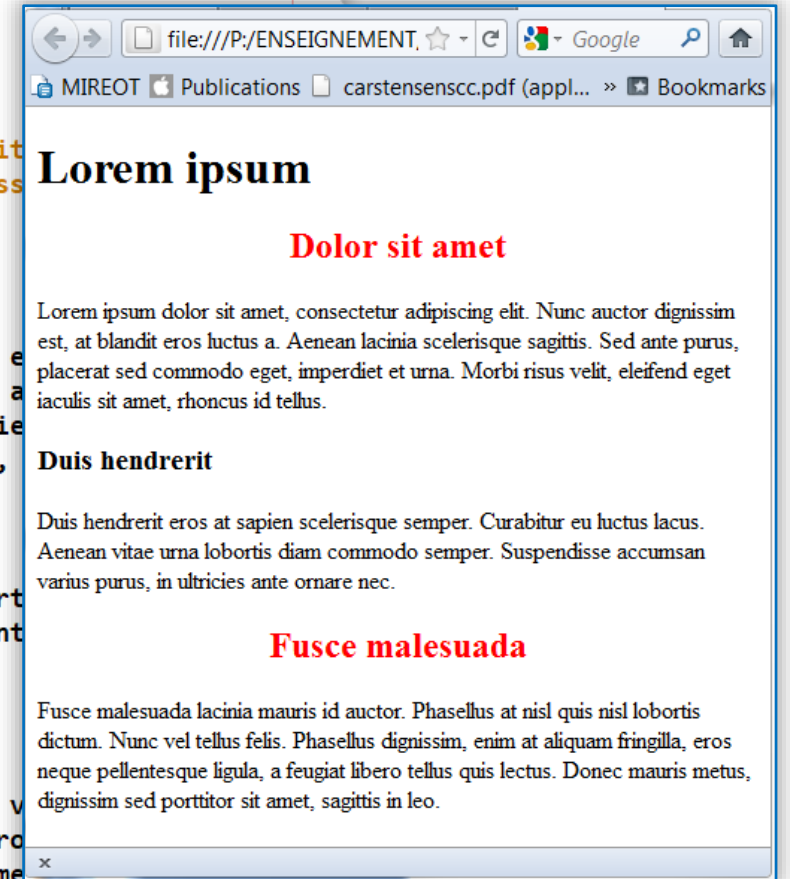
*balise* { *déclaration*; ... }

Le style s'applique à tous les éléments de type **h2**

styles.css

```
h2 {  
  color: red;  
  text-align : center;  
}
```

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title>O5 Sélecteur Balise</title>  
    <meta charset="UTF-8">  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
    <link href="styles.css" rel="stylesheet" type="text/css">  
  </head>  
  <body>  
    <h1>Lorem ipsum</h1>  
    <h2>Dolor sit amet</h2>  
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc auctor dignissim est, at blandit eros luctus a. Aenean lacinia scelerisque sagittis. Sed ante purus, placerat sed commodo eget, imperdiet et urna. Morbi risus velit, eleifend eget iaculis sit amet, rhoncus id tellus.  
    </p>  
    <h3>Duis hendrerit</h3>  
    <p>Duis hendrerit eros at sapien scelerisque semper. Curabitur eu luctus lacus. Aenean vitae urna lobortis diam commodo semper. Suspendisse accumsan varius purus, in ultricies ante  
    </p>  
    <h2>Fusce malesuada</h2>  
    <p>  
    Fusce malesuada lacinia mauris id auctor. Phasellus at nisl quis nisl lobortis dictum. Nunc vel tellus felis. Phasellus dignissim, enim at aliquam fringilla, eros neque pellentesque ligula, a feugiat libero tellus quis lectus. Donec mauris metus, dignissim sed porttitor sit amet, sagittis in leo.  
    </p>  
  </body>  
</html>
```





# Sélecteurs simples

Sélecteur de classe style nommé pouvant être associé à n'importe quel élément à l'aide d'un attribut `class`

forme générale

```
.nomDeClasse { déclaration; ... }
```

The image shows a code editor with two files: `styles.css` and `index.html`. In `styles.css`, a class selector `.renforcé` is defined with `color: red;` and `background-color: yellow;`. In `index.html`, several elements use the `class="renforcé"` attribute, including a paragraph, a heading, and a list item. A browser window on the right shows the rendered page with these elements highlighted in yellow and red. A sticky note explains that the class name should reflect semantics, not presentation.

```
styles.css
.renforcé {
  color: red;
  background-color: yellow;
}
```

```
<!DOCTYPE html>
<html>
  <head>
    <title>06 Sélecteur de classe</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial
    <link href="styles.css" rel="stylesheet" type="text/css"/>
  </head>
  <body>
    <h1>Lorem ipsum</h1>
    <h2>Dolor sit amet</h2>
    <p class="renforcé">Lorem ipsum dolor sit amet, consectetu
    Nunc auctor dignissim est, at blandit eros luctus a. A
    Sed ante purus, placerat sed commodo eget, imperdiet e
    Morbi risus velit, eleifend eget iaculis sit amet, rho
    </p>
    <h3>Duis hendrerit</h3>
    <p>Duis hendrerit eros at sapien scelerisque semper.
    Curabitur eu luctus lacus. Aenean vitae urna lobortis
    Suspendisse accumsan varius purus, in ultricies ante o
    </p>
    <h2 class="renforcé">Fusce malesuada</h2>
    <ul>
      <li>Phasellus at nisl quis nisl lobortis dictum.</li>
      <li class="renforcé">Fusce malesuada lacinia mauris id auctor.</li>
      <li>Phasellus dignissim, enim at aliquam fringilla.</li>
    </ul>
  </body>
</html>
```

Associé à un élément via l'attribut `class`

attribut `class` peut être porté par toute balise → possibilité d'appliquer une même classe de style à des éléments de balises différentes

Le nom de la classe doit refléter la sémantique et non pas la présentation choisie

- Phasellus at nisl quis nisl lobortis dictum.
- **Fusce malesuada lacinia mauris id auctor.**
- Phasellus dignissim, enim at aliquam fringilla.

`.rougeSurJaune`  
`.renforcé`

# Sélecteurs simples

Sélecteur de classe style nommé pouvant être associé à n'importe quel élément à l'aide d'un attribut `class`

forme générale

```
.nomDeClasse { déclaration; ... }
```

possibilité d'appliquer à un même élément plusieurs classes de style différentes

```
styles.css
.renforcé {
  color: red;
  background-color: yellow;
}
.encadré {
  border: solid;
}
```

```
<!DOCTYPE html>
<html>
<head>
  <title>O6 Sélecteur de classe</title>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link href="styles.css" rel="stylesheet" type="text/css">
</head>
<body>
  <h1>Lorem ipsum</h1>
  <h2>Dolor sit amet</h2>
  <p class="renforcé encadré">Lorem ipsum dolor sit amet, Nunc auctor dignissim est, at blandit eros luctus a. Sed ante purus, placerat sed commodo eget, imperdiet Morbi risus velit, eleifend eget iaculis sit amet, r</p>
  <h3>Duis hendrerit</h3>
  <p>Duis hendrerit eros at sapien scelerisque semper. Curabitur eu luctus lacus. Aenean vitae urna lobortis diam commodo semper. Suspendisse accumsan varius purus, in ultricies ante ornare nec.</p>
</body>
</html>
```

Browser preview showing the rendered page with the paragraph highlighted in yellow and having a red border.

# Sélecteurs simples

Sélecteur de classe style nommé pouvant être associé à n'importe quel élément à l'aide d'un attribut `class`

forme générale

```
.nomDeClasse { déclaration; ... }
```

possibilité de restreindre une classe à un type d'élément. Le nom de la classe est préfixé par le nom de la balise

The image shows a code editor with three files: `styles.css`, `index.html`, and a browser preview window titled "06 Sélecteur de classe".

**styles.css**

```
.renforcé {  
  color: red;  
  background-color: yellow;  
}  
  
.encadré {  
  border: solid;  
}  
  
h3.renforcé {  
  font-style: italic;  
  color: blue;  
}
```

**index.html**

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title>06 sélecteur de classe</title>  
    <meta charset="UTF-8">  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
    <link href="styles.css" rel="stylesheet" type="text/css">  
  </head>  
  <body>  
    <h1>Lorem ipsum</h1>  
    <h2>Dolor sit amet</h2>  
    <p class="renforcé encadré">Lorem ipsum dolor sit amet, Nunc auctor dignissim est, at blandit eros luctus a. Sed ante purus, placerat sed commodo eget, imperdiet Morbi risus velit, eleifend eget iaculis sit amet, Duis hendrerit</p>  
    <h3 class="renforcé">Duis hendrerit</h3>  
    <p>Duis hendrerit eros at sapien scelerisque semper. Curabitur eu luctus lacus. Aenean vitae urna lobortis diam commodo semper. Suspendisse accumsan varius purus, in ultricies ante</p>  
    <h2 class="renforcé">Fusce malesuada</h2>  
  </body>  
</html>
```

**Browser Preview (file:///P:/ENSEIGNEME)**

- h1** Lorem ipsum
- h2** Dolor sit amet
- p** Lorem ipsum dolor sit amet, Nunc auctor dignissim est, at blandit eros luctus a. Sed ante purus, placerat sed commodo eget, imperdiet Morbi risus velit, eleifend eget iaculis sit amet, Duis hendrerit
- h3** Duis hendrerit
- p** Duis hendrerit eros at sapien scelerisque semper. Curabitur eu luctus lacus. Aenean vitae urna lobortis diam commodo semper. Suspendisse accumsan varius purus, in ultricies ante
- h2** Fusce malesuada
- ul**
  - Phasellus at nisl quis nisl lobortis dictum.
  - Fusce malesuada lacinia mauris id auctor.
  - Phasellus dignissim, enim at aliquam fringilla.

cette règle de style ne s'applique qu'aux éléments **h3** ayant un attribut `class` avec la valeur **renforcé**

# Sélecteurs simples

Sélecteur d'identificateur style associé à un élément unique identifié par un attribut **id**

forme générale

**#identificateur** { **déclaration;** ... }

- attribut *id* désignant l'élément à styler et défini dans sa balise ouvrante
- doit être unique dans le document HTML

styles.css

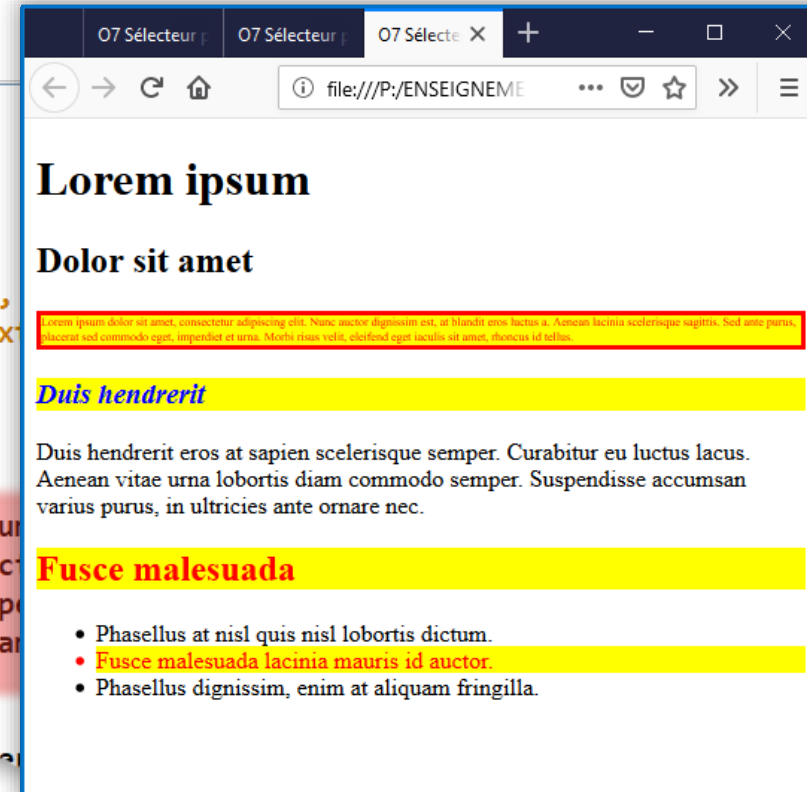
```
.renforcé {
  color: red;
  background-color: yellow;
}

.encadré {
  border: solid;
}

h3.renforcé {
  font-style: italic;
  color: blue;
}

#résumé {
  font-size : 0.8em;
}
```

```
<!DOCTYPE html>
<html>
  <head>
    <title>07 sélecteur par identifiant</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
    <link href="styles.css" rel="stylesheet" type="text/css">
  </head>
  <body>
    <h1>Lorem ipsum</h1>
    <h2>Dolor sit amet</h2>
    <p id="résumé" class="renforcé encadré">Lorem ipsum
    Nunc auctor dignissim est, at blandit eros luctus.
    Sed ante purus, placerat sed commodo eget, imperdiet et urna.
    Morbi risus velit, eleifend eget iaculis sit amet, rhoncus id tellus.
    </p>
    <h3 class="renforcé">Duis hendrerit</h3>
    <p>Duis hendrerit eros at sapien scelerisque semper. Curabitur eu luctus lacus.
    Aenean vitae urna lobortis diam commodo semper. Suspendisse accumsan
    varius purus, in ultricies ante ornare nec.
    </p>
    <ul>
      <li>Phasellus at nisl quis nisl lobortis dictum.</li>
      <li>Fusce malesuada lacinia mauris id auctor.</li>
      <li>Phasellus dignissim, enim at aliquam fringilla.</li>
    </ul>
  </body>
</html>
```



- renforcement : utilisable par plusieurs éléments → sélecteur de classe
- résumé : un seul résumé dans le document → sélecteur d'identificateur

# Groupement de sélecteurs

Groupement de sélecteurs permet d'associer une même règle de style à plusieurs sélecteurs

forme générale

```
selecteur1 , selecteur2 , ...  
{ déclaration; ... }
```

styles.html

```
h1, strong, #dernierP {  
  color: red;  
}
```

remarque : les sélecteurs ne sont pas nécessairement des sélecteurs simples

index.html

```
<!DOCTYPE html>  
<html lang="en">  
  
  <head>  
    <meta charset="UTF-8">  
    <meta http-equiv="X-UA-Compatible" content="IE=edge">  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
    <title>Document</title>  
    <link rel="stylesheet" href="styles.css">  
  </head>  
  <body>  
    <h1>Lorem ipsum dolor sit</h1>  
    <p>Amet consectetur adipisicing elit. Perspiciatis maiores, ipsam quae  
    tempore sit voluptate iste similique mollitia! Quis veniam, quod  
    <strong>temporibus corrupti error</strong> nostrum dicta cupiditate magnam  
    quos similique dicta consectetur fugit? Molestias, officia corporis?  
  </p>  
    <h1>Atque quibusdam perspiciatis</h1>  
    <h2>Provident esse odio</h2>  
    <p id="dernierP">sequi provident eveniet voluptatibus harum quae  
    asperiores magni consequuntur harum id hic minima ipsum quod blanditiis  
    distinctio? Cupiditate exercitationem provident ad voluptatum, ac  
    suscipit inventore ipsa amet! Aspernatur magni eaque ducimus praesentium  
  </p>  
  </body>  
</html>
```

## Lorem ipsum dolor sit

Amet consectetur adipisicing elit. Perspiciatis maiores, ipsam quaeat nulla, neque suscipit, animi aliquam tempore sit voluptate iste similique mollitia! Quis veniam, quod ea quae dignissimos dolore illum saepe **temporibus corrupti error** nostrum dicta cupiditate magnam laborum. Libero ducimus id unde soluta exercitationem quos similique dicta consectetur fugit? Molestias, officia corporis?

## Atque quibusdam perspiciatis

### Provident esse odio

sequi provident eveniet voluptatibus harum quaeat nobis. Ab, nam fugiat! Ea laboriosam voluptates, sapiente, asperiores magni consequuntur harum id hic minima ipsum quod blanditiis praesentium inventore minus suscipit ex distinctio? Cupiditate exercitationem provident ad voluptatum, accusantium error, atque maxime tempore nam suscipit inventore ipsa amet! Aspernatur magni eaque ducimus praesentium harum rem molestiae blanditiis

# Limites des sélecteurs simple

- problème : comment associer un style à des éléments en fonction de leur situation dans l'arborescence ?

ex : particulariser les liens hypertexte de division de navigation(par exemple leur donner une couleur de fond vert clair)

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <nav>
    <ul>
      <li><a href="#">Accueil</a></li>
      <li><a href="cv.html">CV</a></li>
      <li><a href="photos.html">Photos</a></li>
    </ul>
  </nav>
  <section>
    <p>
      Lorem, ipsum dolor sit amet consectetur adipisicing elit. Ducimus ipsum quasi soluta
      accusantium vitae doloremque. Ullam dolor voluptatem vitae neque odio est nostrum,
      dolorum beatae.
    </p>
    <p>Quelques liens intéressants</p>
    <ul>
      <li><a href="http://www.foo.com">Lien 1</a></li>
      <li><a href="http://www.bidon.fr">Lien 2</a></li>
    </ul>
  </section>
</body>
</html>
```

- [Accueil](#)
- [CV](#)
- [Photos](#)

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Ducimus ipsum quasi soluta accusantium vitae doloremque. Ullam dolor voluptatem vitae neque odio est nostrum, dolorum beatae.

Quelques liens intéressants

- [Lien 1](#)
- [Lien 2](#)



- [Accueil](#)
- [CV](#)
- [Photos](#)

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Ducimus ipsum quasi soluta accusantium vitae doloremque. Ullam dolor voluptatem vitae neque odio est nostrum, dolorum beatae.

Quelques liens intéressants

- [Lien 1](#)
- [Lien 2](#)

# Limites des sélecteurs simple

- problème : comment associer un style à des éléments en fonction de leur situation dans l'arborescence ?

ex : particulariser les liens hypertexte de division de navigation(par exemple leur donner une couleur de fond vert clair)

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <nav>
    <ul>
      <li><a href="#" class="navigationLink">Accueil</a></li>
      <li><a href="cv.html" class="navigationLink">CV</a></li>
      <li><a href="photos.html" class="navigationLink">Photos</a></li>
    </ul>
  </nav>
  <section>
    <p>
      Lorem, ipsum dolor sit amet consectetur adipisicing elit. Ducimus ipsum quasi soluta
      accusantium vitae doloremque. Ullam dolor voluptatem vitae neque odio est nostrum,
      dolorum beatae.
    </p>
    <p>Quelques liens interessants</p>
    <ul>
      <li><a href="http://www.foo.com">Lien 1</a></li>
      <li><a href="http://www.bidon.fr">Lien 2</a></li>
    </ul>
  </section>
</body>
</html>
```

avec les styles simples

- définir un classe pour ces liens

```
.navigationLink {
  background-color:lightgreen;
}
```
- ajouter un attribut **class="navigationLink"** à chacun des éléments **a** concernés



Lourd et potentiellement source d'erreurs

- [Accueil](#)
- [CV](#)
- [Photos](#)

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Ducimus ipsum quasi soluta accusantium vitae doloremque. Ullam dolor voluptatem vitae neque odio est nostrum, dolorum beatae.

Quelques liens interessants

- [Lien 1](#)
- [Lien 2](#)

# Les différents types de sélecteurs

*sélecteur* { *propriété: valeur ; ...* }

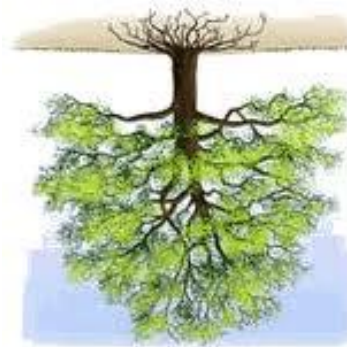
- Les sélecteurs sont utilisés pour sélectionner (trouver) les éléments HTML que l'on veut styler
- Les sélecteurs peuvent être regroupés selon 5 catégories
  - Sélecteurs simples
    - sélectionnent des éléments selon leur type, leur id ou leur classe
  - **Sélecteurs combinés**
    - sélectionnent des éléments selon leur situation dans l'arbre défini par l'imbrication des éléments
  - Sélecteurs de pseudo-classe
    - sélectionnent des éléments en fonction de l'état dans lequel ils se trouvent
  - Sélecteurs de pseudo-élément
    - sélectionnent et stylent une partie d'un élément
  - Sélecteurs d'attributs
    - sélectionnent un élément selon la présence ou la valeur d'un attribut



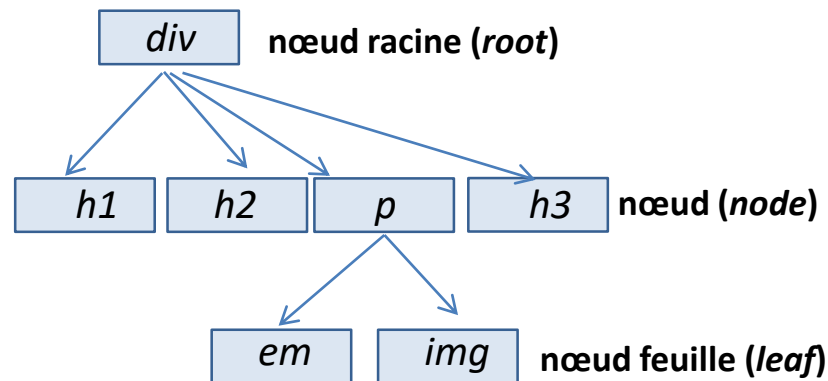
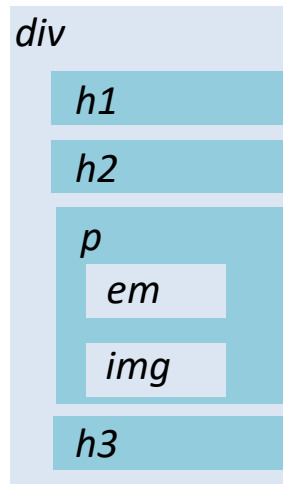
# Sélecteurs combinés

- rappel : arbre des éléments et relations entre éléments

```
<div>  
  <h1>titre</h1>  
  <h2>un sous titre</h2>  
  <p>un paragraphe avec  
    <em>des mots importants</em>  
    pour terminer avec une image  
    .  
  </p>  
  <h3>titre niveau 3</h3>  
</div>
```



Arbre DOM (Document Object Model)



relation *enfant de (child)* :

**p** est enfant de **div**

**em** est enfant de **p**

relation *descendant de* :

un nœud **x** est descendant d'un nœud **y** si **x** est enfant de **y** ou est enfant d'un enfant de **y**

**p** est descendant de **div**

**em** est descendant de **p** et **div**

relation *1<sup>er</sup> frère suivant (adjacent sibling)* :

**p** est le 1<sup>er</sup> frère suivant de **h2**

**img** est le 1<sup>er</sup> frère suivant de **em**

relation *frère suivant (general sibling)* :

un nœud **x** est frère suivant d'un nœud **y** si **x** est 1<sup>er</sup> frère de **y** ou est 1<sup>er</sup> frère d'un frère suivant de **y**

**p** est le frère suivant de **h1** et **h2**

**img** est le frère suivant de **em**

# Sélecteurs combinés

- forme générale

```
selecteur1 combinateur selecteur2 { déclaration; ... }
```

- cible (sélectionne) parmi les éléments sélectionnés par *selecteur*<sub>2</sub>  
uniquement ceux qui ont une relation définie par le combinateur avec un élément sélectionné par *selecteur*<sub>1</sub>
- 4 combinateurs possibles
  - (space) : descendant
  - **>** : descendant direct ou enfant (*child*)
  - **~** : 1<sup>er</sup> frère suivant (*adjacent sibling*)
  - **+** : frère suivant (*general sibling*)

# Sélecteurs combinés

- problème : comment associer un style à des éléments en fonction de leur situation dans l'arborescence ?

ex : particulariser les liens hypertexte de division de navigation(par exemple leur donner une couleur de fond vert clair)

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <nav>
    <ul>
      <li><a href="#">Accueil</a></li>
      <li><a href="cv.html">CV</a></li>
      <li><a href="photos.html">Photos</a></li>
    </ul>
  </nav>
  <section>
    <p>
      Lorem, ipsum dolor sit amet consectetur adipisicing elit. Ducimus ipsum quasi soluta
      accusantium vitae doloremque. Ullam dolor voluptatem vitae neque odio est nostrum,
      dolorum beatae.
    </p>
    <p>Quelques liens intéressants</p>
    <ul>
      <li><a href="http://www.foo.com">Lien 1</a></li>
      <li><a href="http://www.bidon.fr">Lien 2</a></li>
    </ul>
  </section>
</body>
</html>
```

en utilisant combinateur *descendant*

- définir un classe pour les liens situé dans un élément **nav**

```
nav a {
  background-color:lightgreen;
}
```



Facilité d'utilisation

- [Accueil](#)
- [CV](#)
- [Photos](#)

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Ducimus ipsum quasi soluta accusantium vitae doloremque. Ullam dolor voluptatem vitae neque odio est nostrum, dolorum beatae.

Quelques liens intéressants

- [Lien 1](#)
- [Lien 2](#)

# Sélecteurs combinés

Possibilité de combiner des sélecteurs descendants

**sélecteur<sub>1</sub> sélecteur<sub>2</sub> ... sélecteur<sub>n</sub> { déclaration ; ... }**

s'applique à tout élément désigné par *sélecteur<sub>n</sub>* qui descend (directement ou indirectement) d'un élément désigné par *sélecteur<sub>n-1</sub>* ... qui descend d'un élément désigné par *sélecteur<sub>2</sub>* qui descend d'un élément désigné par *sélecteur<sub>1</sub>*

```
<link rel="stylesheet" href="styles_1.css">
</head>
<body>
  <nav>
    <ul>
      <li><a href="#">Accueil</a></li>
      <li><a href="cv.html">CV</a></li>
      <li><a href="photos.html">Photos</a></li>
    </ul>
  </nav>
  <section id="section1">
    <p>
      Lorem ipsum dolor sit amet, consectetur adipisicing elit. Distinctio esse
      voluptatibus hic, <a href="http://www.test.com">perspiciatis</a> atque maxime
      eligendi molestiae quidem?
    </p>
    <ul>
      <li><a href="http://www.foo.com">Lien 1</a></li>
      <li><a href="http://www.bidon.fr">Lien 2</a></li>
    </ul>
  </section>
  <section class="emphase">
    <p>
      Lorem, ipsum dolor sit amet consectetur adipisicing elit. Ducimus ipsum quasi soluta
      accusantium vitae doloremque. Ullam dolor voluptatem vitae neque odio est nostrum,
      dolorum beatae.
    </p>
    <p>Quelques liens intéressants</p>
    <ul>
      <li><a href="http://www.foo.com">Lien 1</a></li>
      <li><a href="http://www.bidon.fr">Lien 2</a></li>
    </ul>
  </section>
</body>
</html>
```

```
nav a {
  background-color: lightgreen;
}
#section1 ul a {
  background-color: yellow;
}
.emphase {
  font-style: italic;
}
.emphase ul a {
  background-color: pink;
}
```

- [Accueil](#)
- [CV](#)
- [Photos](#)

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Distinctio esse voluptatibus hic, [perspiciatis](#) atque maxime eligendi molestiae quidem?

- [Lien 1](#)
- [Lien 2](#)

*Lorem, ipsum dolor sit amet consectetur adipisicing elit. Ducimus ipsum quasi soluta accusantium vitae doloremque. Ullam dolor voluptatem vitae neque odio est nostrum, dolorum beatae.*

*Quelques liens intéressants*

- [Lien 1](#)
- [Lien 2](#)

# Sélecteurs combinés

Sélecteur de nœuds enfants

**sélecteur<sub>1</sub> > sélecteur<sub>2</sub> { déclaration ; déclaration }**

s'applique à tout élément désigné par sélecteur<sub>2</sub> dont le *parent* direct est un élément désigné par sélecteur<sub>1</sub>

Valeur hexadécimale pour spécifier une couleur

```
<body>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipiscing elit.
    <a href="#">Nulla</a> aliquet volutpat ante, nec mollis risus porttitor id.
  </p>
  <div>
    Donec rutrum fermentum consectetur.
    <ul>
      <li>Integer pellentesque sem semper elit gravida luctus.</li>
      <li>Sed ac <a href="#">lacus tortor</a>, a ullamcorper purus.</li>
    </ul>
    Sed in dui <a href="#">in lacus</a> porttitor posuere.
  </div>
  <div id="div1">
    Nam condimentum <a href="#">lectus</a> accumsan lectus tincidunt pharetra.
    <ul>
      <li>Integer pellentesque
        <ul>
          <li>sem <a href="#">semper</a></li>
          <li>elit</li>
        </ul>
        <a href="#">gravida</a> luctus.</li>
      <li>Sed ac <a href="#">lacus tortor</a>, a ullamcorper purus.</li>
    </ul>
  </div>
</body>
```

```
div > a {
  background-color: #F0A8FF;
}
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. [Nulla](#) aliquet volutpat ante, nec mollis risus porttitor id.

Donec rutrum fermentum consectetur.

- Integer pellentesque sem semper elit gravida luctus.
- Sed ac [lacus tortor](#), a ullamcorper purus.

Sed in dui [in lacus](#) porttitor posuere.

Nam condimentum [lectus](#) accumsan lectus tincidunt pharetra.

- Integer pellentesque
  - sem semper
  - elitgravida luctus.
- Sed ac [lacus tortor](#), a ullamcorper purus.

# Sélecteurs combinés

Sélecteur de nœuds enfants

***sélecteur<sub>1</sub> > sélecteur<sub>2</sub> { déclaration ; déclaration }***

s'applique à un élément désigné par sélecteur<sub>2</sub> dont le parent **direct** est un élément désigné par sélecteur<sub>1</sub>

```
<body>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipiscing elit.
    <a href="#">Nulla</a> aliquet volutpat ante, nec mollis risus porttitor id.
  </p>
  <div>
    Donec rutrum fermentum consectetur.
    <ul>
      <li>Integer pellentesque sem semper elit gravida luctus.</li>
      <li>Sed ac <a href="#">lacus tortor</a>, a ullamcorper purus.</li>
    </ul>
    Sed in dui <a href="#">in lacus</a> porttitor posuere.
  </div>
  <div id="div1">
    Nam condimentum <a href="#">lectus</a> accumsan lectus tincidunt pharetra.
    <ul>
      <li>Integer pellentesque
        <ul>
          <li>sem <a href="#">semper</a></li>
          <li>elit</li>
        </ul>
        <a href="#">gravida</a> luctus.</li>
      <li>Sed ac <a href="#">lacus tortor</a>, a ullamcorper purus.</li>
    </ul>
  </div>
</body>
```

```
#div1 > a {
  background-color: #F0A8FF;
}
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. [Nulla](#) aliquet volutpat ante, nec mollis risus porttitor id.

Donec rutrum fermentum consectetur.

- Integer pellentesque sem semper elit gravida luctus.
- Sed ac [lacus tortor](#), a ullamcorper purus.

Sed in dui [in lacus](#) porttitor posuere.

Nam condimentum [lectus](#) accumsan lectus tincidunt pharetra.

- Integer pellentesque
  - sem [semper](#)
  - elit[gravida](#) luctus.
- Sed ac [lacus tortor](#), a ullamcorper purus.

# Sélecteurs combinés

## Combinaisons de sélecteurs

```
<body>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipiscing elit.
    <a href="#">Nulla</a> aliquet volutpat ante, nec mollis risus porttitor id.
  </p>
  <div>
    Donec rutrum fermentum consectetur.
    <ul>
      <li>Integer pellentesque sem semper elit gravida luctus.</li>
      <li>Sed ac <a href="#">lacus tortor</a>, a ullamcorper purus.</li>
    </ul>
    Sed in dui <a href="#">in lacus</a> porttitor posuere.
  </div>
  <div id="div1">
    Nam condimentum <a href="#">lectus</a> accumsan lectus tincidunt pharetra.
    <ul>
      <li>Integer pellentesque
        <ul>
          <li>sem <a href="#">semper</a></li>
          <li>elit</li>
        </ul>
        <a href="#">gravida</a> luctus.</li>
      <li>Sed ac <a href="#">lacus tortor</a>, a ullamcorper purus.</li>
    </ul>
  </div>
</body>
```

identificateur descendant Enfant (*child*)

```
#div1 li > a {
  background-color: #F0A8FF;
}
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. [Nulla](#) aliquet volutpat ante, nec mollis risus porttitor id.

Donec rutrum fermentum consectetur.

- Integer pellentesque sem semper elit gravida luctus.
- Sed ac [lacus tortor](#), a ullamcorper purus.

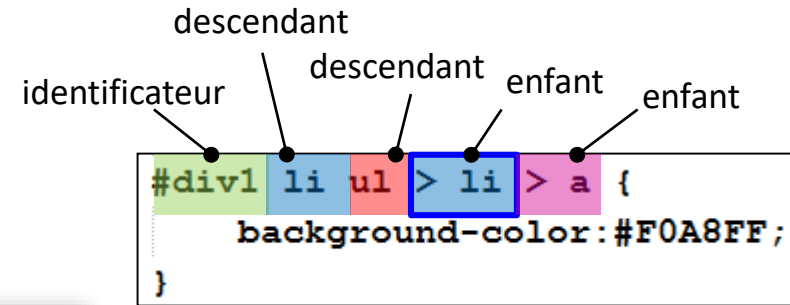
Sed in dui [in lacus](#) porttitor posuere.

Nam condimentum [lectus](#) accumsan lectus tincidunt pharetra.

- Integer pellentesque
  - sem [semper](#)
  - elit[gravida](#) luctus.
- Sed ac [lacus tortor](#), a ullamcorper purus.

# Sélecteurs combinés

## Combinaisons de sélecteurs



```
<body>  
  <p>  
    Lorem ipsum dolor sit amet, consectetur adipiscing elit.  
    <a href="#">Nulla</a> aliquet volutpat ante, nec mollis risus porttitor id.  
  </p>  
  <div>  
    Donec rutrum fermentum consectetur.  
    <ul>  
      <li>Integer pellentesque sem semper elit gravida luctus.</li>  
      <li>Sed ac <a href="#">lacus tortor</a>, a ullamcorper purus.</li>  
    </ul>  
    Sed in dui <a href="#">in lacus</a> porttitor posuere.  
  </div>  
  <div id="div1">  
    Nam condimentum <a href="#">lectus</a> accumsan lectus tincidunt pharetra.  
    <ul>  
      <li>Integer pellentesque  
        <ul>  
          <li>sem <a href="#">semper</a></li>  
          <li>elit</li>  
        </ul>  
      <li><a href="#">gravida</a> luctus.</li>  
      <li>Sed ac <a href="#">lacus tortor</a>, a ullamcorper purus.</li>  
    </ul>  
  </div>  
</body>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. [Nulla](#) aliquet volutpat ante, nec mollis risus porttitor id.

Donec rutrum fermentum consectetur.

- Integer pellentesque sem semper elit gravida luctus.
- Sed ac [lacus tortor](#), a ullamcorper purus.

Sed in dui [in lacus](#) porttitor posuere.

Nam condimentum [lectus](#) accumsan lectus tincidunt pharetra.

- Integer pellentesque
  - sem [semper](#)
  - elit[gravida](#) luctus.
- Sed ac [lacus tortor](#), a ullamcorper purus.



# Sélecteurs combinés

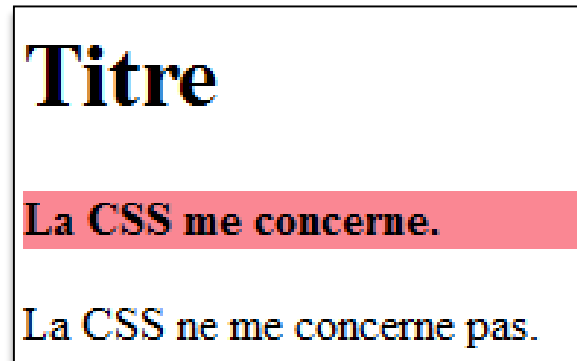
Sélecteur de nœuds adjacents

**sélecteur<sub>1</sub> + sélecteur<sub>2</sub> { déclaration ; déclaration }**

s'applique à tout élément désigné par *sélecteur<sub>2</sub>* dont le frère directement précédent dans le flux du balisage de la page HTML est un élément désigné par *sélecteur<sub>1</sub>*

```
<body>
  <h1>Titre</h1>
  <p>La CSS me concerne.</p>
  <p>La CSS ne me concerne pas.</p>
</body>
```

```
h1 + p {
  font-weight:bold;
  background-color:#FA8793;
}
```



# Sélecteurs combinés

Sélecteur de nœuds frères

**sélecteur1 ~ sélecteur2 { déclaration ; déclaration }**

s'applique à tout élément désigné par *sélecteur<sub>2</sub>* dont un frère précédent dans le flux du balisage de la page HTML est un élément désigné par *sélecteur<sub>1</sub>*

```
<body>
  <h1>Titre niveau 1</h1>
  <h2>Titre niveau 2</h2>
  <p>La CSS me concerne.</p>
  <div>
    <h2>Titre niveau 2</h2>
    <p>La CSS ne me concerne pas.</p>
    <h1>Titre niveau 1</h1>
  </div>
</body>
```

```
h1 ~ p {
  font-weight:bold;
  background-color:#FA8793;
}
```

**Titre niveau 1**

**Titre niveau 2**

La CSS me concerne.

**Titre niveau 2**

La CSS ne me concerne pas.

**Titre niveau 1**

# Les différents types de sélecteurs

*sélecteur* { *propriété: valeur ; ...* }

- Les sélecteurs sont utilisés pour sélectionner (trouver) les éléments HTML que l'on veut styler
- Les sélecteurs peuvent être regroupés selon 5 catégories
  - Sélecteurs simples
    - sélectionnent des éléments selon leur type, leur id ou leur classe
  - Sélecteurs combinés
    - sélectionnent des éléments selon leur situation dans l'arbre défini par l'imbrication des éléments
  - Sélecteurs de pseudo-classe
    - sélectionnent des éléments en fonction de l'état dans lequel ils se trouvent
  - Sélecteurs de pseudo-élément
    - sélectionnent et stylent une partie d'un élément
  - Sélecteurs d'attributs
    - sélectionnent un élément selon la présence ou la valeur d'un attribut

# Sélecteurs de pseudo-classes

- pseudo-classes
  - portent sur des éléments existants dans le code source du document et auxquels on peut accéder par des caractéristiques autres que leur nom, attribut ou contenu.

**sélecteur**:pseudoClasse { declaration; ... }

- structurelles : **:first-child**, **:last-child**, **:nth-child(n)** ....
- d'ancres **:link** et **:visited**
- dynamiques **:hover**, **:active** et **:focus**

The image illustrates the use of CSS pseudo-class selectors through three overlapping windows:

- HTML Code Window:** Shows the source code for a page titled "Changement de la couleur de fond quand la souris passe sur l'un des items de cette lsite." It includes a style block that imports "css/styleHover.css" and a list of three items: "Première ligne", "Deuxième ligne", and "Troisième ligne". A link is also present: "Et maintenant, un [lien](#) pour illustrer le hover".
- CSS Rules Window:** Shows two CSS rules:
  - `li:hover { background-color : aqua; }`
  - `a:hover { text-decoration : none; color: red; }`
- Browser Window (Top):** Shows the rendered page. The first list item, "Première ligne", is highlighted with a cyan background, demonstrating the `li:hover` rule. The link is underlined and red, demonstrating the `a:hover` rule.
- Browser Window (Bottom):** Shows the rendered page with the same content, but the link is underlined and red, demonstrating the `a` rule.

# Les différents types de sélecteurs

*sélecteur* { propriété: valeur ; ... }

- Les sélecteurs sont utilisés pour sélectionner (trouver) les éléments HTML que l'on veut styler
- Les sélecteurs peuvent être regroupés selon 5 catégories
  - Sélecteurs simples
    - sélectionnent des éléments selon leur type, leur id ou leur classe
  - Sélecteurs combinés
    - sélectionnent des éléments selon leur situation dans l'arbre défini par l'imbrication des éléments
  - Sélecteurs de pseudo-classe
    - sélectionnent des éléments en fonction de l'état dans lequel ils se trouvent
  - **Sélecteurs de pseudo-élément**
    - sélectionnent et styler une partie d'un élément
  - Sélecteurs d'attributs
    - sélectionnent un élément selon la présence ou la valeur d'un attribut

# Sélecteurs de pseudo éléments

- pseudo-éléments

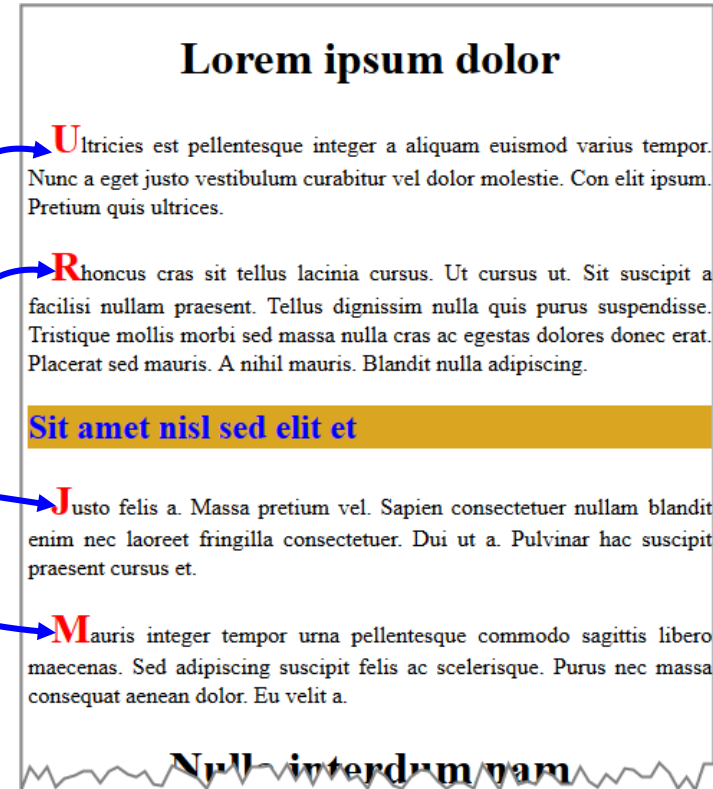
- permettent de donner un style à un contenu n'apparaissant pas dans le code source du document comme étant un élément.

sélecteur **::pseudoElt** { déclaration; ... }

- exemples

- **::first-line**
- **::first-letter**
- **::before**
- **::after**
- **::selection**
- ...

```
h2 {  
  color: blue;  
  background-color: goldenrod;  
}  
  
h1 {  
  text-align: center;  
}  
  
p {  
  text-indent: 16px;  
  text-align: justify;  
}  
  
p::first-letter {  
  font-size: 1.8em;  
  color: red;  
  font-weight: bolder;  
}
```



# Les différents types de sélecteurs

*sélecteur* { *propriété: valeur ; ...* }

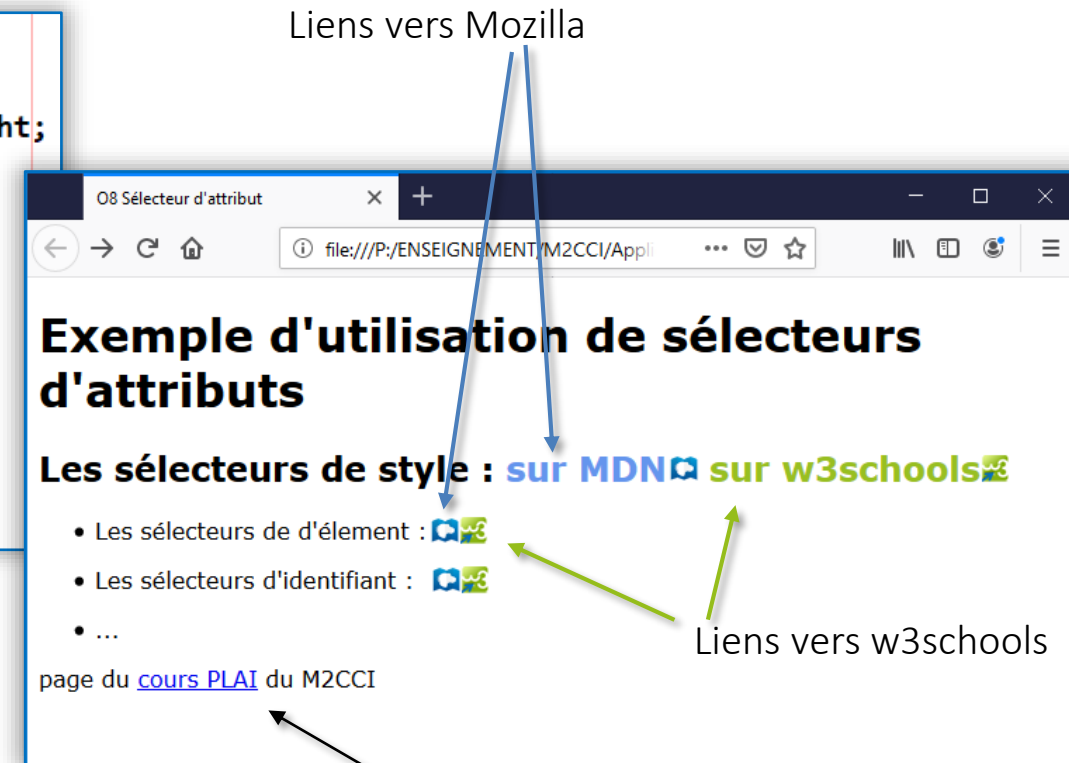
- Les sélecteurs sont utilisés pour sélectionner (trouver) les éléments HTML que l'on veut styler
- Les sélecteurs peuvent être regroupés selon 5 catégories
  - Sélecteurs simples
    - sélectionnent des éléments selon leur type, leur id ou leur classe
  - Sélecteurs combinés
    - sélectionnent des éléments selon leur situation dans l'arbre défini par l'imbrication des éléments
  - Sélecteurs de pseudo-classe
    - sélectionnent des éléments en fonction de l'état dans lequel ils se trouvent
  - Sélecteurs de pseudo-élément
    - sélectionnent et stylent une partie d'un élément
  - **Sélecteurs d'attributs**
    - sélectionnent un élément selon la présence ou la valeur d'un attribut

# Sélecteurs d'attribut

- permettent de cibler un élément selon la présence d'un attribut ou selon la valeur donnée d'un attribut
  - Exemple : particulariser les liens selon leur destination

```
a[href ^= 'https://www.w3schools'] {
  color:#98BF21;
  background: transparent url('./images/w3schools.png') no-repeat center right;
  padding-right: 20px;
  text-decoration: none;
}

a[href ^= 'https://developer.mozilla.org'] {
  color:cornflowerblue;
  background: transparent url('./images/MDN.png') no-repeat center right;
  padding-right: 22px;
  text-decoration: none;
}
```



Pour plus de détails

[https://developer.mozilla.org/fr/docs/Web/CSS/Sélecteurs d attribut](https://developer.mozilla.org/fr/docs/Web/CSS/Sélecteurs_d_attribut)



# Sélecteur Universel

Sélecteur universel (\*) sélectionne tous les éléments de la page

## index.html

```
<!DOCTYPE html>
<html lang="en">

  <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <h1>Lorem ipsum dolor sit</h1>
    <p>Amet consectetur adipisicing elit. Perspiciatis maiores, ipsam quaerat nulla, neque susci
    tempore sit voluptate iste similique mollitia! Quis veniam, quod ea quae dignissimos dol
    <strong>temporibus corrupti error</strong> nostrum dicta cupiditate magnam laborum. Libe
    quos similique dicta consectetur fugit? Molestias, officia corporis?
    </p>
    <h1>Atque quibusdam perspiciatis</h1>
    <h2>Provident esse odio</h2>
    <p>sequi provident eveniet voluptatibus harum quaerat nobis. Ab, nam fugiat! Ea laboriosam
    asperiores magni consequuntur harum id hic minima ipsum quod blanditiis praesentium inve
    distinctio? Cupiditate exercitationem provident ad voluptatum, accusantium error, atque
    suscipit inventore ipsa amet! Aspernatur magni eaque ducimus praesentium harum rem moles
    </p>
  </body>
</html>
```

## styles.html

```
* {
  border-style: solid;
  border-color: red;
  border-width: 1px;
  padding: 5px;
  margin: 10px;
}
```

## Lorem ipsum dolor sit

Amet consectetur adipisicing elit. Perspiciatis maiores, ipsam quaerat nulla, neque suscipit, animi aliquam tempore sit voluptate iste similique mollitia! Quis veniam, quod ea quae dignissimos dolore illum saepe **temporibus corrupti error** nostrum dicta cupiditate magnam laborum. Libero ducimus id unde soluta exercitationem quos similique dicta consectetur fugit? Molestias, officia corporis?

## Atque quibusdam perspiciatis

### Provident esse odio

sequi provident eveniet voluptatibus harum quaerat nobis. Ab, nam fugiat! Ea laboriosam voluptates, sapiente, asperiores magni consequuntur harum id hic minima ipsum quod blanditiis praesentium inventore minus suscipit ex distinctio? Cupiditate exercitationem provident ad voluptatum, accusantium error, atque maxime tempore nam suscipit inventore ipsa amet! Aspernatur magni eaque ducimus praesentium harum rem molestiae blanditiis

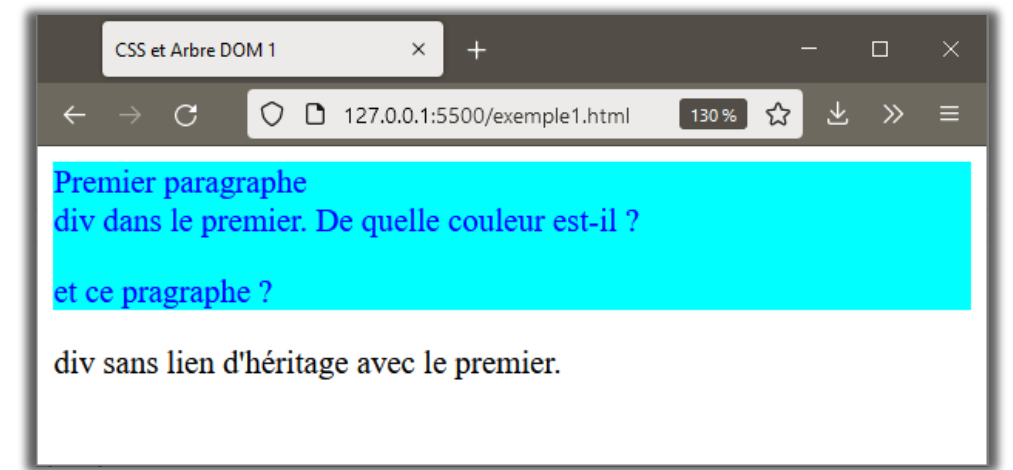
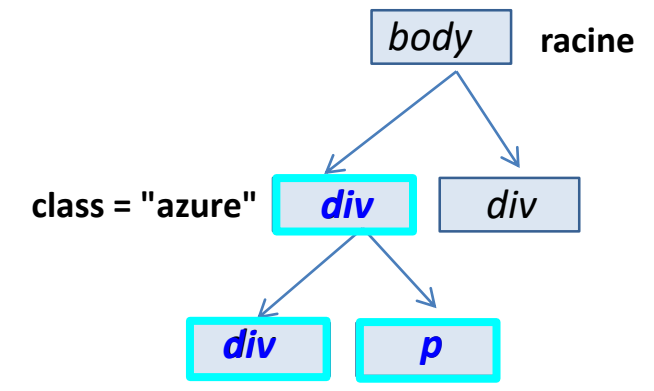
# Propagation des styles dans l'arbre des éléments

- propagation des propriétés\* d'un élément aux éléments descendants

\* En fait cela ne concerne pas toutes les propriétés (ex. border pas de propagation)

```
styles1.css x
css > styles1.css > ...
1 .azure {
2     background-color: aqua;
3     color: blue;
4 }
```

```
exemple1.html x
exemple1.html > ...
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <title>CSS et Arbre DOM 1</title>
8     <link rel="stylesheet" href="css/styles1.css">
9 </head>
10 <body>
11     <div class="azure">
12         Premier paragraphe
13         <div>
14             div dans le premier. De quelle couleur est-il ?
15         </div>
16         <p>
17             et ce pragraphe ?
18         </p>
19     </div>
20     <div>
21         div sans lien d'héritage avec le premier.
22     </div>
23 </body>
24 </html>
```

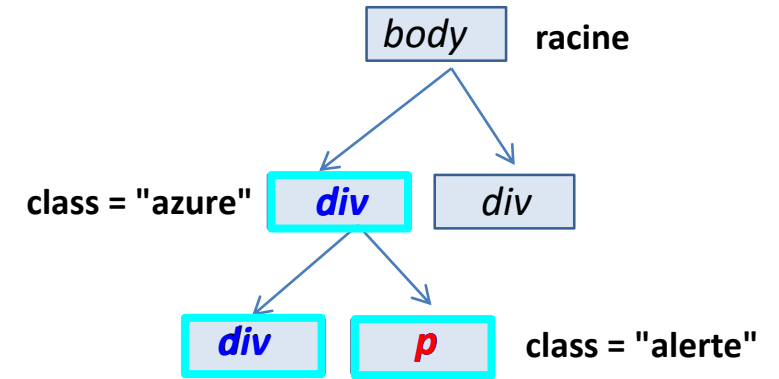


# Propagation des styles dans l'arbre des éléments

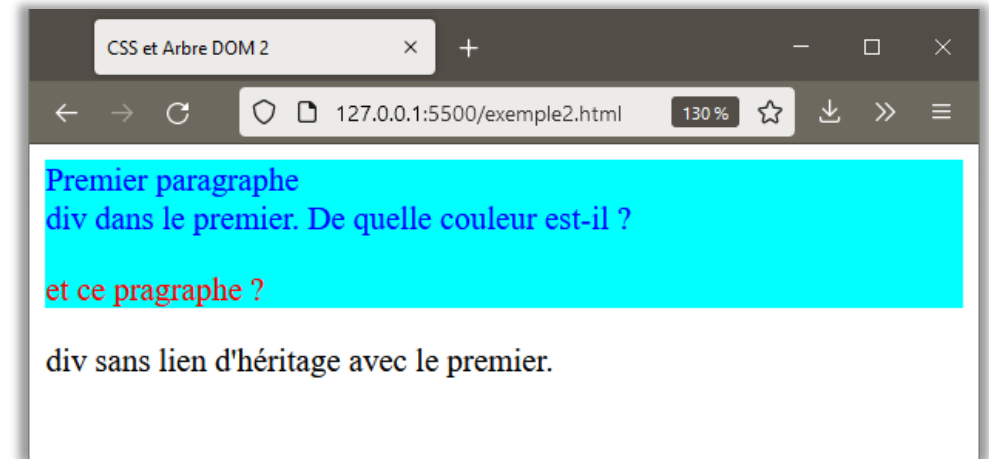
- propagation des propriétés\* d'un élément aux éléments descendants

\* En fait cela ne concerne pas toutes les propriétés (ex. border pas de propagation)

```
styles2.css x
css > styles2.css > ...
1  .azure {
2      background-color: aqua;
3      color: blue;
4  }
5
6  .alerte {
7      color: red;
8  }
```



```
exemple2.html x
exemple2.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>CSS et Arbre DOM 2</title>
8      <link rel="stylesheet" href="css/styles2.css">
9  </head>
10 <body>
11     <div class="azure">
12         Premier paragraphe
13         <div>
14             div dans le premier. De quelle couleur est-il ?
15         </div>
16         <p class="alerte">
17             et ce pragraphe ?
18         </p>
19     </div>
20     <div>
21         div sans lien d'héritage avec le premier.
22     </div>
23 </body>
24 </html>
```



# Cascade

- Que se passe-t-il si plusieurs règles de style rentrent en conflit (concernent une même propriété) ?
- 1<sup>er</sup> cas : une même classe CSS définit 2 fois avec une propriété en conflit

```
* CSS
1 .title {
2   color: blue;
3 }
4
5 .title {
6   color: red;
7 }
```

Quand tout est égal (même sélecteur, même propriété) la dernière déclaration écrase les précédentes

```
HTML
1 <h1 class="title">Hello World</h1>
2
3 <h1 class="title">Bye Bye World</h1>
4
```

**Hello World**  
**Bye Bye World**

# Cascade

- Que se passe-t-il si plusieurs règles de style rentrent en conflit (concernent une même propriété) ?
- 1<sup>er</sup> cas : une même classe CSS définit 2 fois avec une propriété en conflit

```
* CSS
1 .title {
2   color: blue;
3   border: solid 1px blue;
4 }
5 .title {
6   color: red;
7 }
```

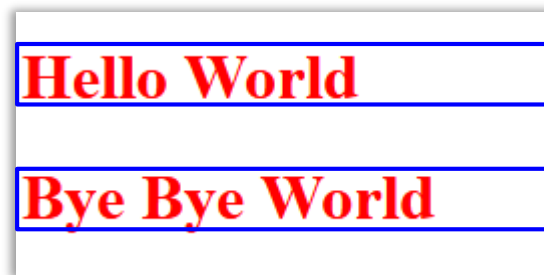
Quand tout est égal (même sélecteur, même propriété) la dernière déclaration écrase les précédentes



Les conflits sont résolus propriété par propriété

- La déclaration `border: solid 1px blue` de la 1<sup>ère</sup> règle est conservée
- La déclaration `color: blue` de la 1<sup>ère</sup> règle est remplacée celle de la 2<sup>ème</sup> règle

```
HTML
1 <h1 class="title">Hello World</h1>
2
3 <h1 class="title">Bye Bye World</h1>
4
```



# Cascade

- 2<sup>ème</sup> cas : deux classes CSS différentes avec une propriété en conflit

```
* CSS
2 ▾ .surligne {
3   color : red;
4   background-color : yellow;
5 }
6
7 ▾ .title {
8   color : blue;
9 }
```

```
HTML
1 ▾ <h1 class="title">Hellow World 1</h1>
2 ▾ <h1 class="surligne">Hello World 2</h1>
3 ▾ <h1 class="title surligne">Hello World 3</h1>
```

le style appliqué est **title**

**Hellow World 1**

# Cascade

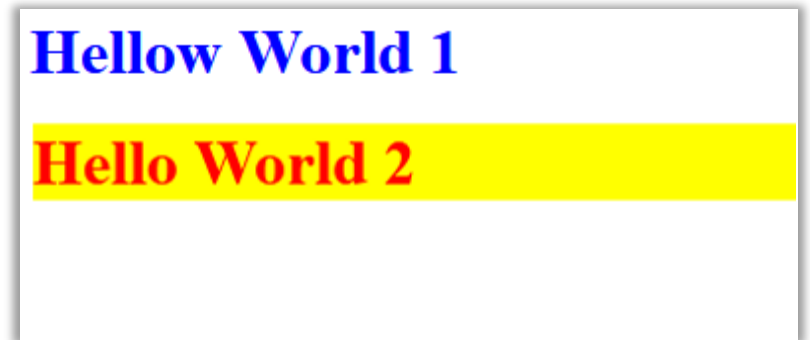
- 2<sup>ème</sup> cas : deux classes CSS différentes avec une propriété en conflit

```
* CSS
2 ▾ .surligne {
3   color : red;
4   background-color : yellow;
5 }
6
7 ▾ .title {
8   color : blue;
9 }
```

```
HTML
1 ▾ <h1 class="title">Hello World 1</h1>
2 ▾ <h1 class="surligne">Hello World 2</h1>
3 ▾ <h1 class="title surligne">Hello World 3</h1>
```

le style appliqué est **title**

le style appliqué est **surligne**



# Cascade

- 2<sup>ème</sup> cas : deux classes CSS différentes avec une propriété en conflit

```
* CSS
2 ▾ .surligne {
3   color : red;
4   background-color : yellow;
5 }
6
7 ▾ .title {
8   color : blue;
9 }
```

la propriété **color** est définie dans deux règles dont les sélecteurs (**.surligne** et **.title**) ont la même **spécificité** : c'est la dernière déclaration qui écrase les autres

```
HTML
1 ▾ <h1 class="title">Hellow World 1</h1>
2 ▾ <h1 class="surligne">Hello World 2</h1>
3 ▾ <h1 class="title surligne">Hello World 3</h1>
```

le style appliqué est **title**  
le style appliqué est **surligne**  
les 2 règles de style sont appliquées



? comment elles se combinent ?  
quel est la valeur de la propriété color ?



# Cascade

- 3<sup>ème</sup> cas : conflit entre plusieurs classes

```
* CSS
1 ▾ .surligne {
2   color : red;
3   background-color : yellow;
4 }
5
6 ▾ .surligne.title {
7   color:green;
8 }
9
10 ▾ .title {
11   color : blue;
12 }
```

```
HTML
1 ▾ <h1 class="title">Hello World 1</h1>
2 ▾ <h1 class="surligne">Hello World 2</h1>
3 ▾ <h1 class="title surligne">Hello World 3</h1>
```

le style appliqué est **title**



# Cascade

- 3<sup>ème</sup> cas : conflit entre plusieurs classes

```
* CSS
1 ▾ .surligne {
2   color : red;
3   background-color : yellow;
4 }
5
6 ▾ .surligne.title {
7   color:green;
8 }
9
10 ▾ .title {
11   color : blue;
12 }
```

```
HTML
1 ▾ <h1 class="title">Hello World 1</h1>
2 ▾ <h1 class="surligne">Hello World 2</h1>
3 ▾ <h1 class="title surligne">Hello World 3</h1>
```

le style appliqué est **title**

le style appliqué est **surligne**

**HelloWorld 1**

**HelloWorld 2**

# Cascade

- 3<sup>ème</sup> cas : conflit entre plusieurs classes

```
* CSS
1 ▾ .surligne {
2   color : red;
3   background-color : yellow;
4 }
5
6 ▾ .surligne.title {
7   color:green;
8 }
9
10 ▾ .title {
11   color : blue;
12 }
```

```
HTML
1 ▾ <h1 class="title">Hello World 1</h1>
2 ▾ <h1 class="surligne">Hello World 2</h1>
3 ▾ <h1 class="title surligne">Hello World 3</h1>
```

le style appliqué est **title**

le style appliqué est **surligne**

les 3 règles de style peuvent être appliquées



comment elles se combinent ?

quel est la valeur de la propriété color ?

**HelloWorld 1**

**HelloWorld 2**

# Cascade

- 3<sup>ème</sup> cas : conflit entre plusieurs classes

```
* CSS
1 ▾ .surligne {
2   color : red;
3   background-color : yellow;
4 }
5
6 ▾ .surligne.title {
7   color:green;
8 }
9
10 ▾ .title {
11  color : blue;
12 }
```

Le sélecteur `.surligne.title` est **plus spécifique** que les autres (il ne sélectionnera que les éléments ayant les deux classes), la propriété `color` de cette règle écrase les déclarations des autres règles

```
HTML
1 ▾ <h1 class="title">Hello World 1</h1>
2 ▾ <h1 class="surligne">Hello World 2</h1>
3 ▾ <h1 class="title surligne">Hello World 3</h1>
```

le style appliqué est `title`  
le style appliqué est `surligne`  
les 3 règles de style peuvent être appliquées

? comment elles se combinent ?  
quel est la valeur de la propriété `color` ?



# Cascade

- calcul de la spécificité d'une règle se style

10

```
* CSS
1 .title {
2   color: green;
3 }
4
5 h1 {
6   color: blue;
7   background-color: yellow;
8 }
```

1

```
HTML
1
2 <h1 class="title">Hello World</h2>
3
```

Le sélecteur de classe `.title` est **plus spécifique** que le sélecteur de balise (tag). La propriété `color` de cette règle écrase celle de la règle `h1`.

De manière générale une règle de style associée à un sélecteur de classe est considéré comme **plus spécifique** qu'une règle de style associée à un sélecteur de balise.

Une manière de déterminer la spécificité et de lui attribuer un poids :

- 10 si sélecteur de classe,
- 1 si sélecteur de balise.

La spécificité la plus grande l'emporte.



# Cascade

- calcul de la spécificité d'une règle de style

11

```
* CSS
1 h1.title {
2   color : blue;
3   background-color: yellow;
4 }
5
10 6 .title {
7   color: green;
8 }
```

10

Le sélecteur **h1.title** est **plus spécifique** que le sélecteur de classe (tag) **.title**.

Spécificité de **.title** = 10 (.title classe)

Spécificité de **h1.title** = 1 (h1 tag) + 10 (.title classe)

```
HTML
1
2 <h1 class="title">Hello World</h2>
3
```



# Cascade

- calcul de la spécificité d'une règle de style

```
* CSS
1 h1.title {
2   color: blue;
3   background-color: yellow;
4 }
5
6
7 .alert.title {
8   color: red;
9 }
10
11 .title {
12   color: green;
13 }
```

11

20

10

Le sélecteur **alert.title** est **plus spécifique** que le sélecteur de **h1.title** lui même plus spécifique que le sélecteur **.title**

Spécificité de **.title** = 10 (**.title** classe)

Spécificité de **h1.title** = 1 (**h1** tag) + 10 (**.title** classe)

Spécificité de **.alert.title** = 10 (**.alert** tag) + 10 (**.title** classe)

```
HTML
1
2 <h1 class="alert title">Hello World</h2>
3
```



# Cascade

- calcul de la spécificité d'une règle de style

```
* CSS
1 #the-title {
2   color: magenta;
3 }
4
5 h1.title {
6   color: blue;
7   background-color: yellow;
8 }
9
10 .alert.title {
11   color: red;
12 }
13
14 .title {
15   color: green;
16 }
```

100

11

20

10

Un sélecteur d'identifiant est plus spécifique que des sélecteurs de classe eux-mêmes plus spécifiques que des sélecteurs de balises

Spécificité de `.title` = 10 (`.title` classe)

Spécificité de `h1.title` = 1 (`h1` tag) + 10 (`.title` classe)

Spécificité de `.alert.title` = 10 (`.alert` tag) + 10 (`.title` classe)

Spécificité de `#the-title` = 100 (`# id`)

```
HTML
1
2 <h1 id="the-title" class="alert title">Hello World</h2>
3
```





# Cascade

- calcul de la spécificité d'une règle de style

```
* CSS
1 #the-title {
2   color: magenta;
3 }
4
5 h1.title {
6   color: blue;
7   background-color: yellow;
8 }
9
10 .alert.title {
11   color: red !important;
12 }
13
14 .title {
15   color: green;
16 }
```

100

11

1000

10

Une propriété avec **!important** est plus spécifique que toutes les autres



**!important**

Évitez de l'utiliser !



**#id**



**.class**



**<tag>**

```
HTML
1
2 <h1 id="the-title" class="alert title">Hello World</h2>
3
```



# Sélecteurs de style - Priorité des règles

- La règle exacte de priorité est la suivante

Si deux règles de style pouvant s'appliquer à un élément ont des déclarations contradictoires, celles de la deuxième (dernière règle rencontrée) remplacent celles de la première (la précédente) sauf si la première a un degré de priorité (c.a.d. de spécificité) supérieur à la deuxième

Règle de calcul de degré de priorité (spécificité) :  $x_1 \ x_2 \ x_3 \ x_4$

- $x_1$  : 1 si style en ligne ou **!important**, 0 sinon
- $x_2$  : nombre d'identifiants (**#xxx**) dans le sélecteur
- $x_3$  : nombre de classes (**.xxx**) dans le sélecteur
- $x_4$  : nombre d'éléments (**tag**) séparés par des espaces dans le sélecteur

Pour plus de détails : [http://www.openweb.eu.org/articles/cascade\\_css/](http://www.openweb.eu.org/articles/cascade_css/)

# Définition des couleurs

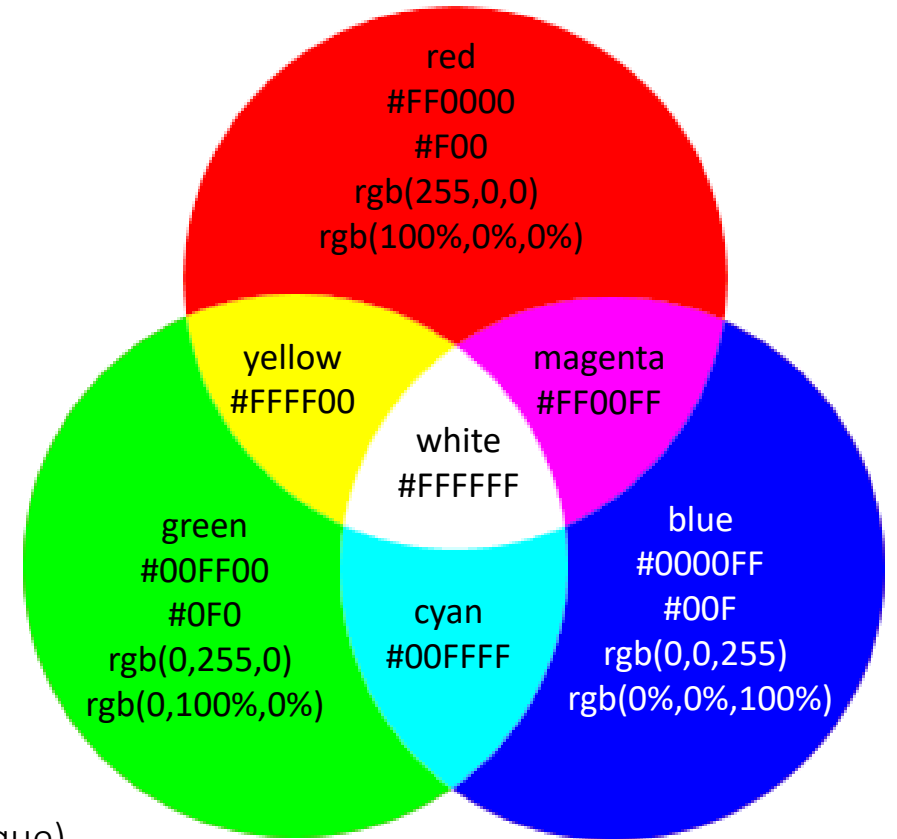
<https://www.w3schools.com/colors/default.asp>

- CSS2

- nom de couleur [https://www.w3schools.com/colors/colors\\_names.asp](https://www.w3schools.com/colors/colors_names.asp)
- synthèse additive à partir des couleurs primaires (Rouge, Vert, Bleu)
  - chaque composante est définie par une valeur entre 0 et 255
  - notation hexadécimale
    - #FFFF00 [https://www.w3schools.com/colors/colors\\_hexadecimal.asp](https://www.w3schools.com/colors/colors_hexadecimal.asp)
  - notation décimale
    - rgb(255,255,0) [https://www.w3schools.com/colors/colors\\_rgb.asp](https://www.w3schools.com/colors/colors_rgb.asp)

- CSS3

- canal alpha (opacité)
  - rgba(r,g,b,a)
  - opacité: nombre entre 0 (totalement transparent) et 1 (totalement opaque)



#9d00ff  
rgb(157,0,255)

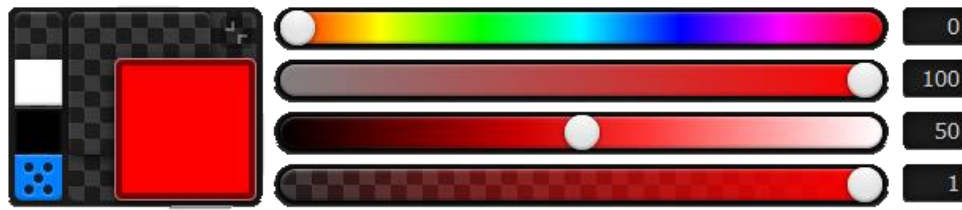
<https://hslpicker.com>

# Définition des couleurs

<https://www.w3schools.com/colors/default.asp>

- CSS3 [https://www.w3schools.com/colors/colors\\_hsl.asp](https://www.w3schools.com/colors/colors_hsl.asp)  
autre modèle que RGB, le modèle HSL (Hue, Saturation, Lightness)
  - `hsl(hue,saturation,lightness)`
  - `hsla(h,s,l,a)` avec canal alpha

<https://hslpicker.com>



**hue** (teinte) définie par un nombre (angle entre 0 et 360) sur la roue des couleurs (color wheel)

**saturation** : dégradé entre gris (0%) et la teinte (100%)

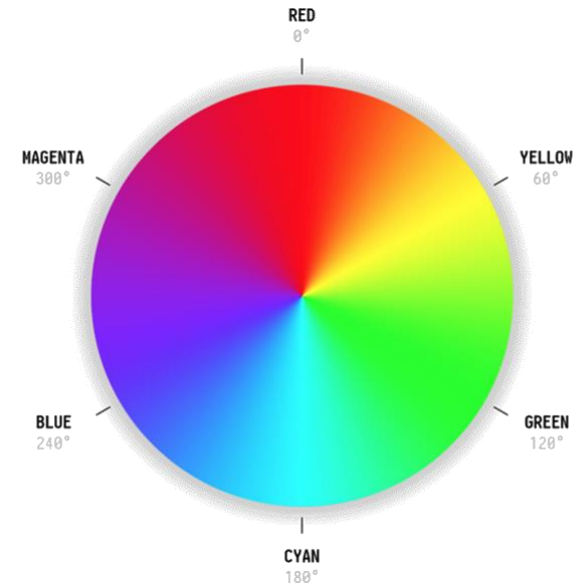
**lightness** (clarté) : dégradé entre noir (0%) et blanc (100%) , 50% correspond à la teinte

**opacité** nombre entre 0 (totalement transparent) et 1 (totalement opaque)

#f00

rgba(255, 0, 0, 1)

hsla(0, 100%, 50%, 1)



- CSS4

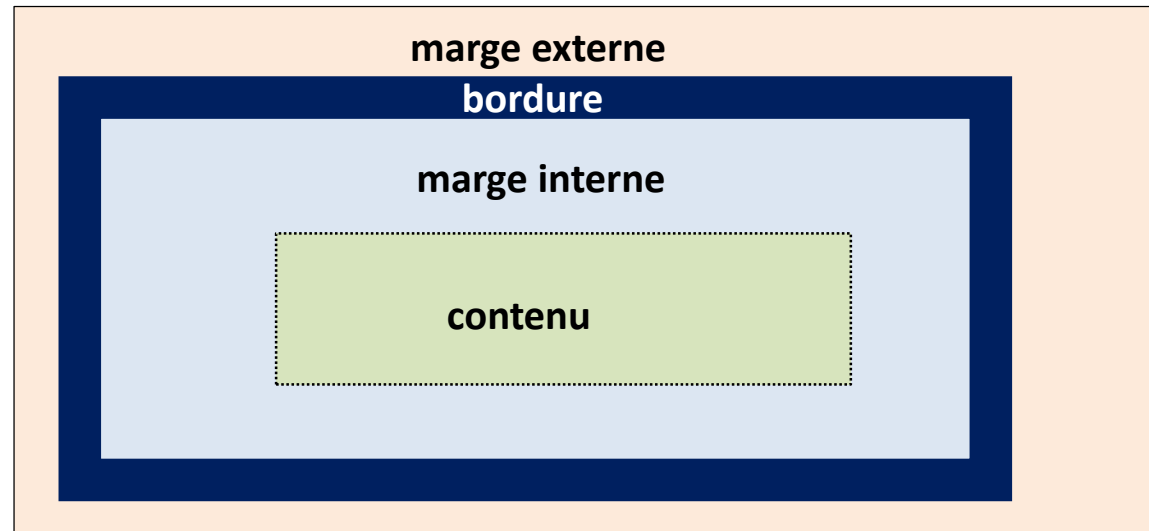
d'autres modèle de couleur en préparation

- CMYK (Cyan, Magenta, Yellow, Black) utilisé pour les imprimantes [https://www.w3schools.com/colors/colors\\_hwb.asp](https://www.w3schools.com/colors/colors_hwb.asp)
- HWB (Hue Whiteness, Blackness) [https://www.w3schools.com/colors/colors\\_cmyk.asp](https://www.w3schools.com/colors/colors_cmyk.asp)

# Définition des couleurs

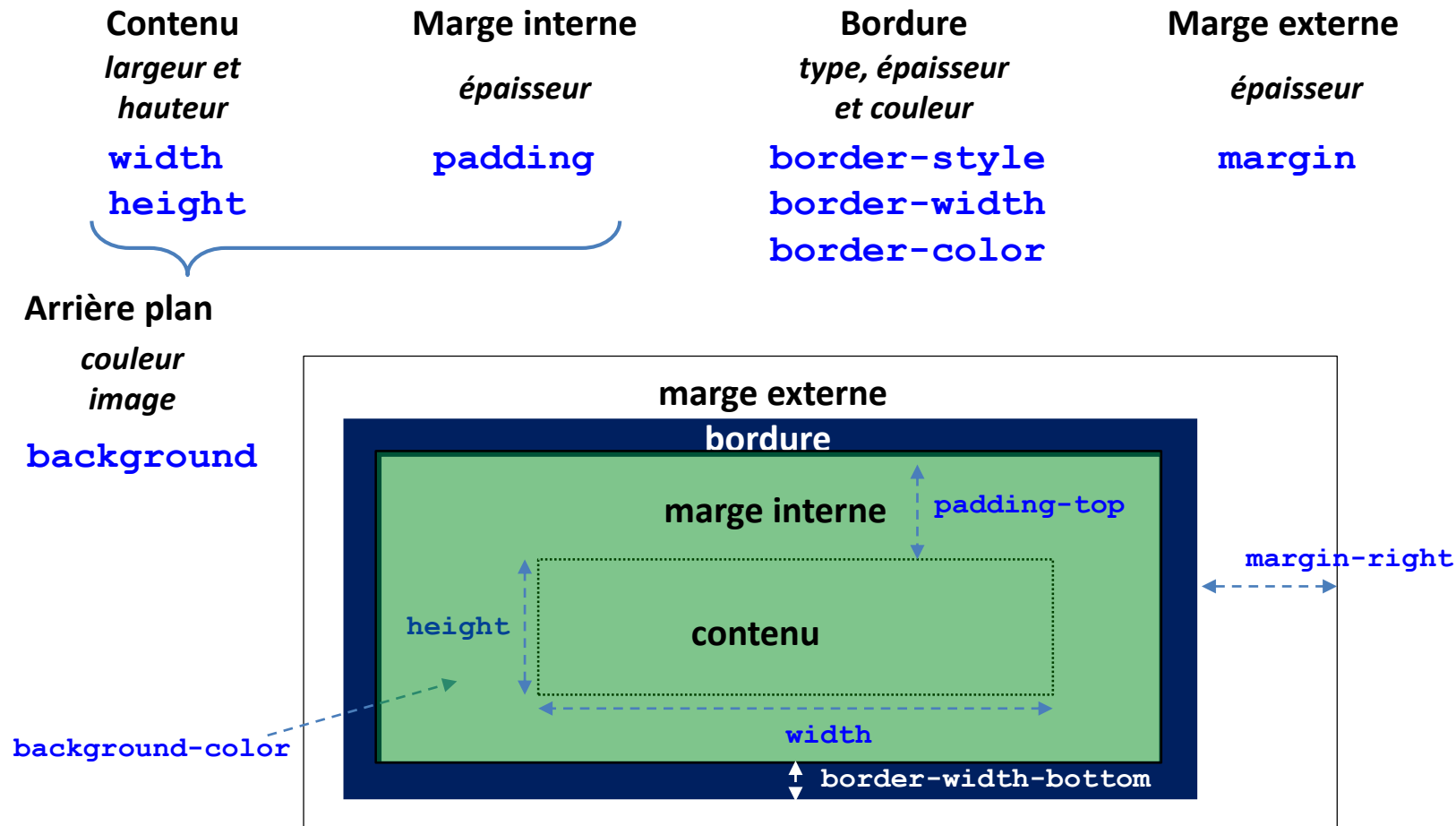
# Modèle de boîte

- boîte rectangulaire matérialisant un élément composée de 3 parties :
  - son contenu (contenu de l'élément : texte, image... autres éléments...)
  - un cadre
    - une bordure
    - un marge interne (marge entre le contenu et la bordure)
  - une marge externe
    - marge entre le cadre et les boîtes matérialisant les éléments adjacents dans le flux

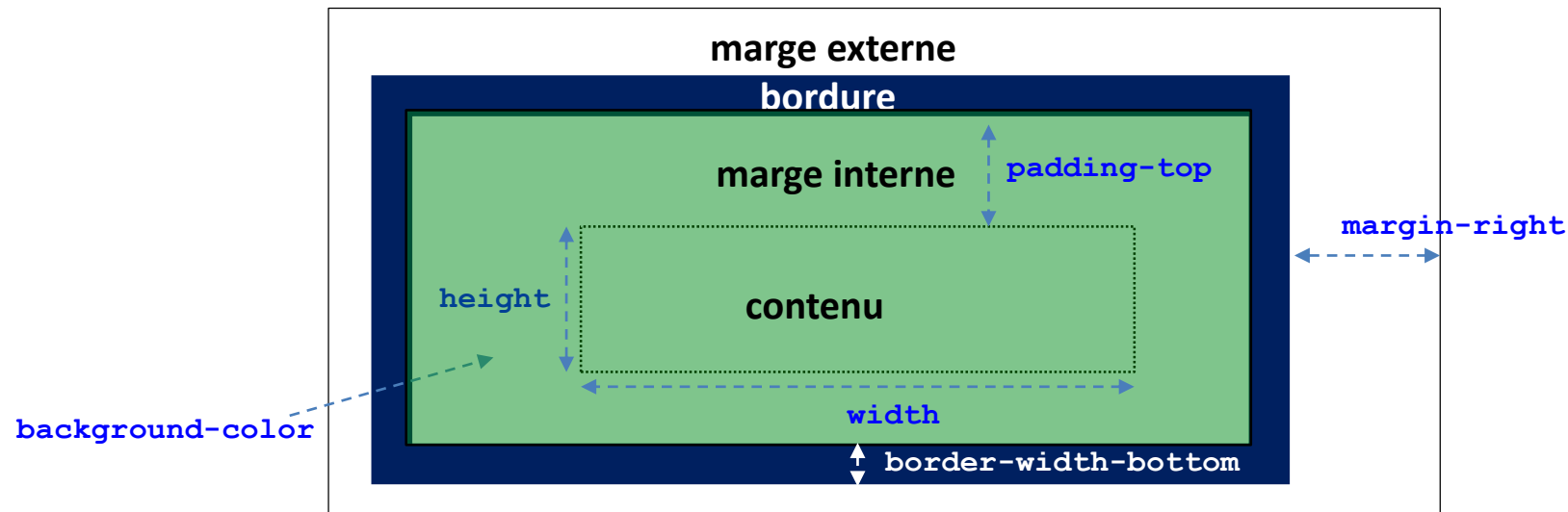


# Modèle de boîte

- La taille du contenu, le cadre et la marge externe d'une boîte peuvent être contrôlés par des propriétés CSS.



- dimensions d'une boîte induites par son contenu + trois périmètres (*margin*, *border*, *padding*).
- *margin*, *border*, *padding* pas obligatoires
  - valeurs par défaut
    - 0 pour éléments en ligne et `<div>`
    - *border* = 0, *margin* et *padding* dépendent du navigateur pour les éléments bloc  
→ fixer dans les feuilles de style CSS pour rendu indépendant du navigateur



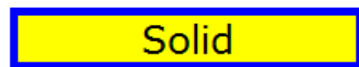
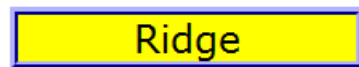
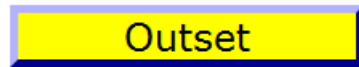
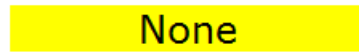
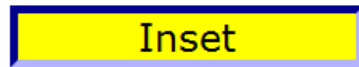
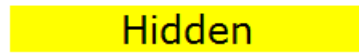
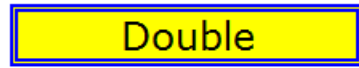
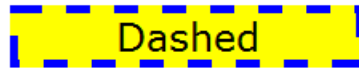


# Modèle de boites

- Bordures et arrière plan
    - Possibilité d'entourer les éléments et de les doter d'un arrière-plan (couleur, image)
    - Propriétés
      - **border-style** style
      - **border-width** épaisseur
      - **border-color** couleur
      - **background-color** couleur arrière plan
      - **background-image** image d'arrière plan
- } applicable à tous les bords ou de chaque côté

# Modèle de boîte

- **border-style**



valeurs possibles

- **border-style:<val1>** tous les côtés
- **border-style:<val1> <val2>**
  - <val1> côtés horizontaux
  - <val2> côtés verticaux
- **border-style:<val1> <val2> <val3>**
  - <val1> côté horizontal haut
  - <val2> côtés verticaux
  - <val3 > côté horizontal bas
- possibilité de spécifier chaque côté
  - **border-left-style**
  - **border-right-style**
  - **border-top-style**
  - **border-bottom-style**

# Modèle de boîte

## Bordures et arrière-plan épaisseur, couleur

- **border-width**

- valeur numérique ex: **border-width: 5px** (5 pixels)

- mots clés



- **border-color**

- valeur numérique

- hexadécimale : **#00ffff** ou **#0ff**

- rgb : **rgb(0, 255, 255)** ou **rgb(0, 100%, 100%)**

- mot clé : **cyan**

- même règles que pour **border-style** pour individualiser chaque côté.

# CSS

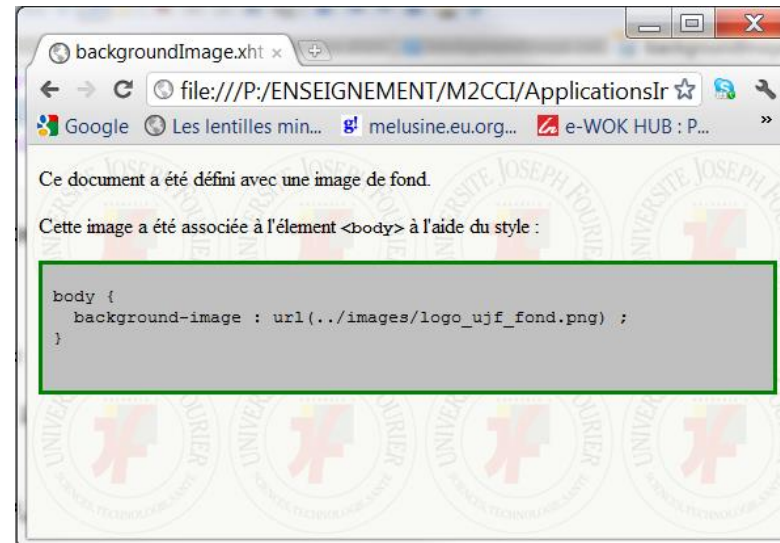
## Bordures et arrière-plan couleur – image arrière-plan

- **background-color:** valeur
  - #00ffff, #0ff, rgb(0,255,255), rgb(0,100%,100%), cyan
- **background-image:** url(*url de l'image*)
  - image locale  
**url(images/logo-ujf.png)** – chemin relatif
  - image distante **url(http://www.monsite.org/images/logo.gif)**

– par défaut l'image  
est répétée en damier



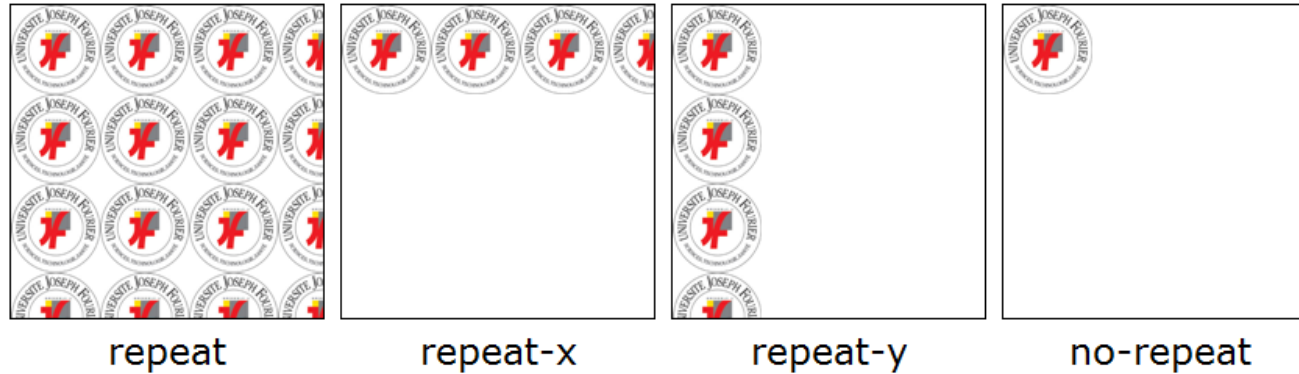
logo\_ujf-fond.png



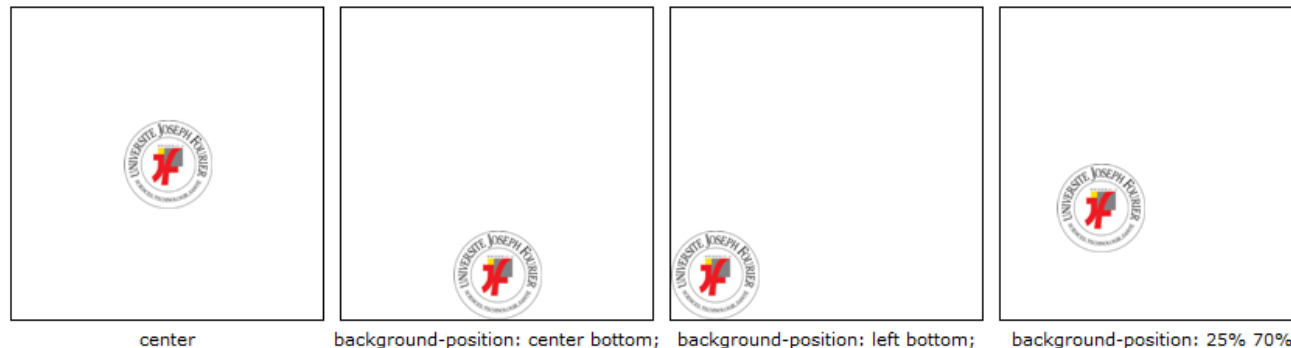
# Modèle de boîte

## Bordures et arrière-plan image arrière-plan

- **background-repeat:** *valeur*
  - contrôle répétition de l'image d'arrière-plan



- **background-position:** *valeur1 valeur2*
  - *valeur1* position horizontale par rapport au bord gauche  
**left, right** ou pourcentage
  - *valeur2* si présente, position verticale par rapport au bord supérieur  
**top, bottom** ou pourcentage



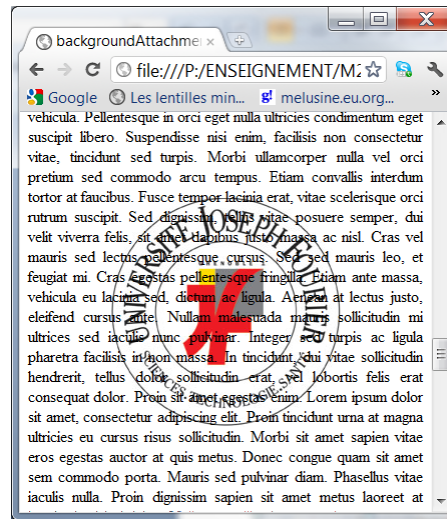
# Modèle de boîte

## Bordures et arrière-plan image arrière-plan

- Par défaut image d'arrière-plan attachée à son élément
  - se déplace avec lui lors d'un scroll
- **background-attachment: fixed**
  - attache l'image à la fenêtre.



```
body {  
  background-image : url("../images/logo_ujf.png");  
  background-position : center ;  
  background-repeat: no-repeat;  
  background-attachment: fixed;  
  text-align: justify;  
}
```



```
body {  
  background-image : url("../images/logo_ujf.png");  
  background-position : center ;  
  background-repeat: no-repeat;  
  background-attachment: scroll;  
  text-align: justify;  
}
```

