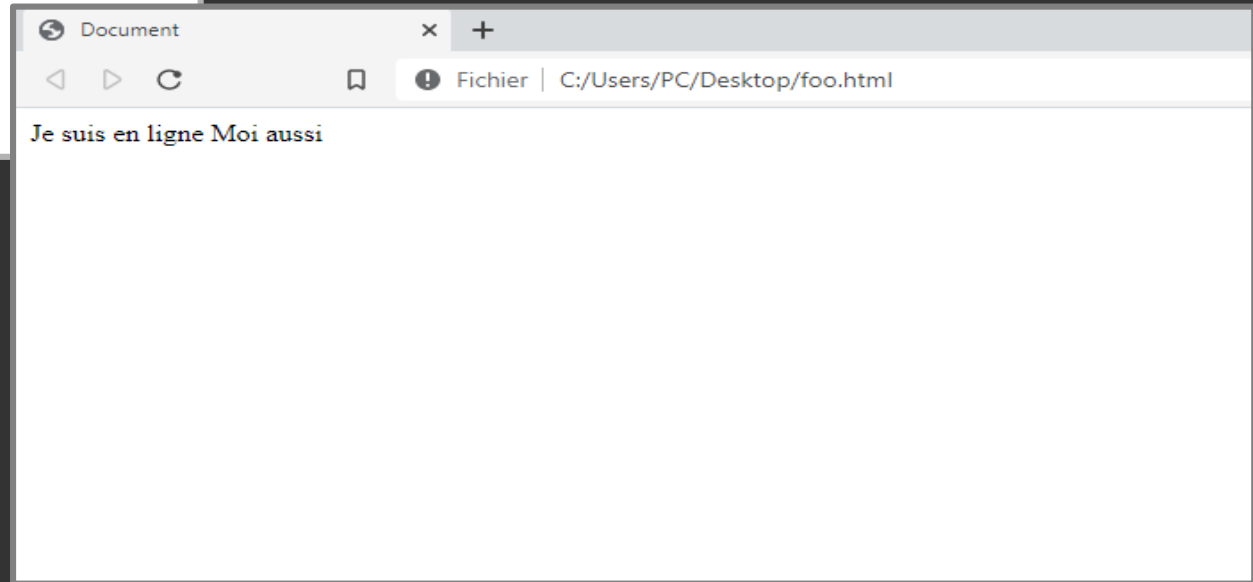
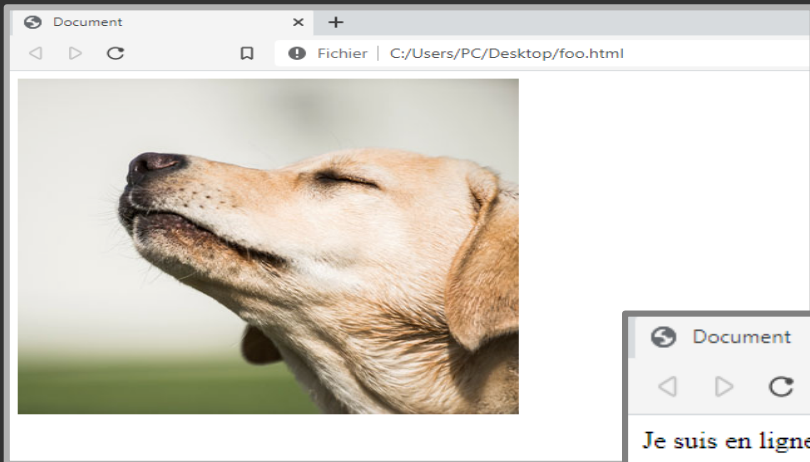


Intégration Web

MMI 1 – TP#2 S1

Danielo **JEAN-LOUIS**
Michele **LINARDI**

Voilà ce que nous avons produit jusqu'à présent



Pas très beau non ?

CSS – Feuilles de style

- Signifie Cascading Style Sheets
 - Feuilles de Style en Cascade
- Décrit la présentation d'une page html

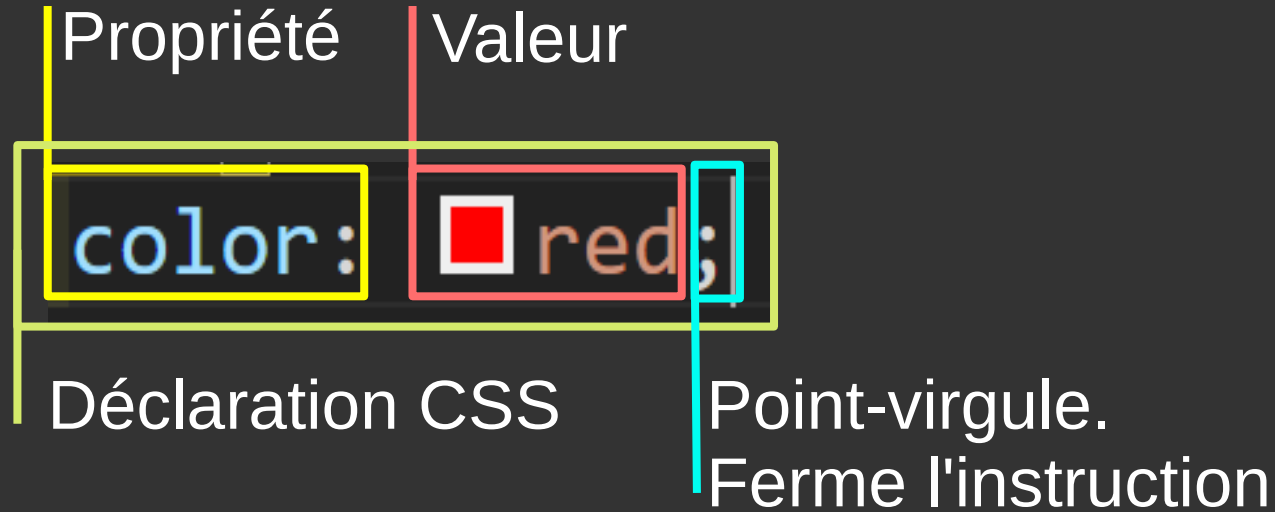
CSS – Feuilles de style

- Version actuelle 3 (CSS3)
- Le langage évolue assez régulièrement
- Stylise le texte via des **déclarations CSS**

CSS – Feuilles de style

- Syntaxe d'une déclaration CSS :
 - **Propriété** : effet que l'on souhaite donner (taille, couleur de texte, espacement)
 - **Valeur** : valeur de la propriété ("pink", 42px, etc.)
 - **Point-virgule (;)** : ferme l'instruction

CSS – Feuilles de style - syntaxe



CSS – Feuilles de style

- Toutes les propriétés possèdent une valeur par défaut, ces valeurs ne sont pas visibles dans le fichier css.
 - Elles sont gérées par le navigateur
- Certaines valeurs ne s'appliquent qu'à certaines propriétés
 - Exemple : La propriété "color" ne peut pas avoir comme valeur "42px"

CSS – Déclaration CSS - valeur

- Il ne doit pas avoir d'espace entre une valeur et son unité

```
1 .mauvais-exemple {  
2   font-size: 2 rem;  
3 }
```



Incorrect :

Il y a un espace entre "2" et "rem", la valeur ne s'appliquera pas

```
1 .bon-exemple {  
2   font-size: 2rem;  
3 }
```



Correct :

Il n'y a pas d'espace entre "2" et "rem", la valeur s'appliquera

CSS – Feuilles de style

- Les navigateurs ne connaissent pas toutes les propriétés
 - Vérifiable grâce au site : <https://caniuse.com/>
- Chaque navigateur a sa façon d'interpréter le CSS, des différences de rendu sont possibles
- Les commentaires s'écrivent entre `/* */`

Source :

- <https://caniuse.com/>

CSS – Feuilles de style

- Le CSS peut s'appliquer :
 - Directement dans la balise html
 - Entre les balises `<style></style>` propre à la page
 - Dans un fichier distant avec l'extension .css

Sources :

- https://developer.mozilla.org/fr/docs/Learn/CSS/First_steps

CSS – Dans la balise html

- Utilisation de l'attribut "style"

```
<p style="color: red; /*font-size: 30px;*/ text-decoration:underline;">  
  Bonjour tout le monde  
</p>
```

La propriété "color" permet de définir la couleur du
texte d'une balise

Bonjour tout le monde

Et il s'affiche en rouge, comme défini dans le code

- La balise html fait office de **sélecteur CSS**

Pratiquons ! - Préparons la SAE 105

Pré-requis :

- Avoir la ressource **en-route-vers-sae-105**

A télécharger ici :

<https://download-directory.github.io/?url=https%3A%2F%2Fgithub.com%2FDanYellow%2Fcours%2Ftree%2Fmain%2Fintegration-web-s1%2Fen-route-vers-sae-105>

CSS – Dans la balise html - Inconvénients

- Maintenabilité
 - Chaque balise doit avoir son propre attribut "style" → Si on doit changer la couleur de **toutes les balises** <p> ça sera très fastidieux
- Lisibilité
 - Une balise avec 5+ déclarations dans l'attribut "style" peut rendre rapidement illisible son fichier HTML s'il y a beaucoup de balises

CSS – Balise <style>

- Peut-être placée dans la balise <head> ou <body> (<head> par convention)
- Souvent vue avec l'attribut "type" et la valeur "text/css"
- Plusieurs occurrences possibles sur la même page de la balise
- Utilise également un système de **sélecteur CSS**

Source :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/style>

CSS – Sélecteur css

- Sert à appliquer une ou plusieurs **déclarations CSS** dans un fichier lié OU entre les balises `<style></style>`
- N'importe quelle balise html peut servir de **sélecteur CSS** et ses **déclarations CSS** s'appliquent à toutes les balises qui ont le même nom

```
p {  
  color: purple;  
  font-size: 25px;  
}
```

Tous les textes entre balises `<p>` de la page auront le texte violet et une taille de texte de 25 px

CSS – Sélecteur CSS

sélecteur CSS (ici de balise ou de type)

commentaire

valeur

p

color: red;

/*font-size: 30px;*/


text-decoration: underline;


}

propriété CSS

déclaration CSS

CSS – Sélecteur CSS vs attribut style

```
p {  
  color:  red;  
  /*font-size: 30px;*/  
  text-decoration: underline;  
}
```

```
<p style='color:  red; /*font-size: 30px;*/ text-decoration:underline;'>  
  Bonjour tout le monde  
</p>
```

Pratiquons ! - Préparons la SAE 105

Pré-requis :

- Avoir la ressource **en-route-vers-sae-105**

A télécharger ici :

<https://download-directory.github.io/?url=https%3A%2F%2Fgithub.com%2FDanYellow%2Fcours%2Ftree%2Fmain%2Fintegration-web-s1%2Fen-route-vers-sae-105>

CSS – Balise <style> - Avantage

- Séparation de la logique
 - La définition du style est séparée du HTML → Moins de confusions dans le code HTML

CSS – Balise `<style>` - Inconvénient

- Maintenabilité
 - Si on a 20 pages, il faudra changer le contenu de la balise `<style>` dans les 20 pages → Fastidieux

CSS – Feuille de style externe

- Extension .css
- Feuille **liée** via la balise <link> dans la balise <head>

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

- L'emplacement du fichier .css est important pour le lier au fichier .html
- Rappelez-vous le précédent TP, la partie sur les chemins relatifs

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/link>

CSS – Feuille de style externe

- Possibilité de mettre plusieurs balises <link> dans la même page
 - Attention : L'ordre des balises compte
- Utilise un système de **sélecteurs CSS**

Source :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/link>

CSS – Sélecteurs CSS

- Permettent de cibler des éléments HTML selon plusieurs critères :
 - Balise (appelé aussi "de type")
 - classe
 - id
 - ...

Source :

- <https://www.pierre-giraud.com/liste-selecteurs-css-utilisation/>

CSS – Sélecteur de balise / type

- Mauvaise pratique :
 - Crée des effets de bord :
 - **Affecte tous les éléments ayant la même balise**

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/Type_selectors

CSS – Sélecteur CSS de classe

- Contrevient à limitation des sélecteurs de balises
 - Rappel : Un sélecteur de balises s'applique à toutes les balises de la page
- Utilisation de l'attribut "class"
 - Attribut global donc utilisable sur n'importe quelle balise html

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/S%C3%A9lecteurs_de_classe

CSS – Sélecteur CSS de classe

- Une même classe peut être utilisée par plusieurs balises HTML
 - Même si elles sont différentes
 - Réutilisable

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/S%C3%A9lecteurs_de_classe

CSS – Sélecteur CSS de classe

- Un nom de classe est arbitraire MAIS il est préférable qu'il soit explicite
 - Vous développez avec d'autres
 - Vous développez pour le vous du futur
- Un nom de classe ne peut pas commencer par un chiffre
 - Mais il peut en contenir

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/S%C3%A9lecteurs_de_classe

CSS – Sélecteur CSS de classe

- Une balise HTML peut avoir plusieurs classes CSS (séparées par un espace)
- Permet un ciblage plus spécifique
- Dans le fichier CSS (ou la balise <style>), on cible une classe en la préfixant d'un point "."

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/S%C3%A9lecteurs_de_classe

CSS – Sélecteur CSS de classe

```
<p class="mon-article urgent">  
    Nouvelle urgente : Ceci est un cours.  
</p>
```

Ici notre balise <p> possède combien de classes ?

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/S%C3%A9lecteurs_de_classe

CSS – Sélecteur CSS de classe - syntaxe

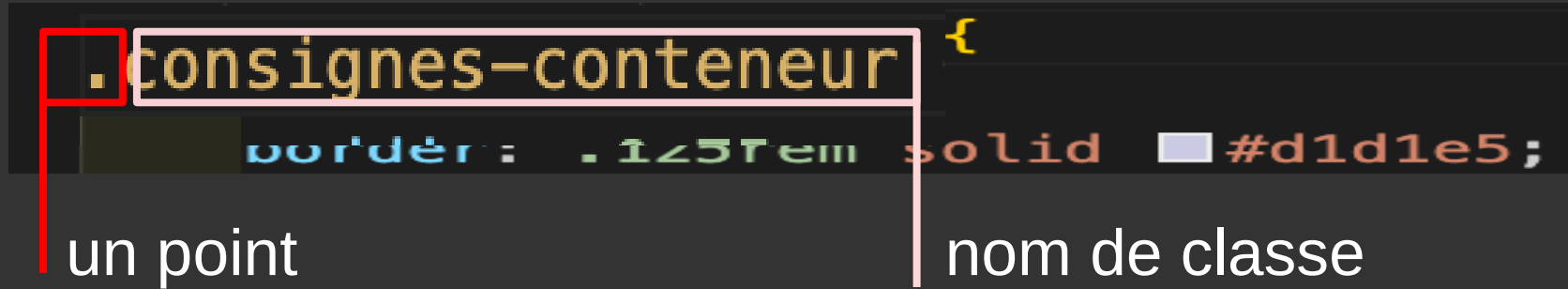
```
.consignes-conteneur {  
    padding: 0.75rem;  
    border: .125rem solid #d1d1e5;  
    margin: 1rem 0;  
}
```

Sélecteur de
classe

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/S%C3%A9lecteurs_de_classe

CSS – Sélecteur CSS de classe - syntaxe



- Sélecteur de classe : "." + "nom-de-classe"
- Sélecteur d'id : "#" + "nom-de-id"
 - Rappel : une valeur d'id par page

Sources :

- https://developer.mozilla.org/fr/docs/Web/CSS/S%C3%A9lecteurs_de_classe
- https://developer.mozilla.org/fr/docs/Learn/Getting_started_with_the_web/CSS_basics

CSS – Sélecteur CSS de classe

- Pas d'espace entre le caractère et son nom
- Les noms de classes sont insensibles à la casse :
 - **.MaClasse** c'est la même chose que **".maclasse"**...

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/S%C3%A9lecteurs_de_classe

CSS – Sélecteur CSS de classe

- ...mais le HTML est sensible à la casse

```
<style>
  .monElement {
    display: flex;
    flex-direction: row;
    justify-content: flex-end;
  }
</style>
<head>
<body>
  <section class="monElement">
    <!-- [...] -->
  </section>
```



Le style ne s'applique pas

```
<style>
  .monelement {
    display: flex;
    flex-direction: row;
    justify-content: flex-end;
  }
</style>
<head>
<body>
  <section class="monelement">
    <!-- [...] -->
  </section>
```



Le style s'applique

CSS – Sélecteur CSS de classe

- Préférez donc une cohérence dans le nommage des classes et id
 - Par convention, les classes/id sont écrites en minuscules (kebab case)
Ex : `.ma-classe` ou `#mon-identifiant`

Sources :

- https://developer.mozilla.org/fr/docs/Web/CSS/S%C3%A9lecteurs_de_classe
- <https://stackoverflow.com/questions/12533926/are-class-names-in-css-selectors-case-sensitive> – anglais
- https://fr.wiktionary.org/wiki/kebab_case
- <https://blog.nicolas.brondin-bernard.com/convention-nommage-kebab-case-snake-case-et-autres/#le-kebab-case>

Pratiquons ! - Préparons la SAE 105

Pré-requis :

- Avoir la ressource **en-route-vers-sae-105**

A télécharger ici :

<https://download-directory.github.io/?url=https%3A%2F%2Fgithub.com%2FDanYellow%2Fcours%2Ftree%2Fmain%2Fintegration-web-s1%2Fen-route-vers-sae-105>

CSS – Feuille de style externe - Avantages

- Polyvalence
 - Possibilité de lier plusieurs fichiers .html aux mêmes fichiers .css
- Cohérence dans le design
 - Toutes les pages html liées aux mêmes fichiers css auront les mêmes designs au niveau de leurs balises

CSS – Feuille de style externe - Avantages

- Maintenabilité
 - Si on change le fichier .css, tous les fichiers html liés à un fichier .css verront les modifications s'appliquer
 - Le style et la structure sont séparés → plus lisible

CSS – Feuille de style externe - Avantages

- Gestion des supports
 - La balise <link>, utilisée pour lier un fichier css, accepte comme attribut "media", ceci permet d'utiliser des fichiers css pour certains supports. Par exemple une imprimante :

```
<link rel="stylesheet"  
      type="text/css"  
      href="print.css" media="print" />
```

La feuille de style "print.css" ne s'applique qu'aux imprimantes grâce à l'attribut "media" et la valeur "print"

CSS – Sélecteur de classe

- Un élément HTML peut avoir plusieurs classes, elles doivent être séparées par un espace :

```
<span class="un texte-important informations">  
  Texte rouge aussi  
</span>
```

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/S%C3%A9lecteurs_de_classe

CSS – Sélecteur de classe

- L'ordre des classes dans l'attribut "class" n'influe en rien sur l'interprétation des déclarations
 - C'est dans les fichiers .css (ou balises <style>) que ceci a son importance

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/S%C3%A9lecteurs_de_classe

CSS – Sélecteur CSS

- Dans un sélecteur CSS, si la classe est préfixée d'une balise HTML, on cible la balise ayant la classe en question

```
p.pistil {  
  color: green;  
}
```

Le style
ne s'applique pas



```
<span class="pistil">
```

Texte non vert

```
</span>
```

Le style
s'applique



```
<p class="pistil">
```

Texte vert

```
</p>
```

CSS – Sélecteur CSS

- Cette règle s'applique également avec une combinaison de classes...

```
.pistil.fleur {  
  color: green;  
}
```

Le style
ne s'applique pas



``
Texte non vert
``

Le style
s'applique



`<p class="pistil fleur">`
Texte vert
`</p>`

CSS – Sélecteur CSS

- ...et plein d'autres cas :
 - classe + id : **#mon-id.ma-classe**
 - attribut + classe : **[src].ma-classe**
 - classe + classe + classe : **.cl1.cl2.cl3**
- A noter que l'ordre ne compte pas

.cl1.cl2.cl3 = .cl3.cl1.cl3

Dans les deux cas nous ciblerons les éléments HTML ayant les trois classes

Pratiquons ensemble ! - Sélecteurs CSS

Pré-requis :

- Avoir la ressource **ressources/css-selecteurs**


A télécharger ici :

<https://download-directory.github.io/?url=https%3A%2F%2Fgithub.com%2FDanYellow%2Fcours%2Ftree%2Fmain%2Fintegration-web-s1%2Ftravaux-pratiques%2Fnumero-3%2Fressources%2Fcss-selecteurs>

Connectez-vous avec votre compte Teams/Office 365 pour télécharger la ressource

CSS – Sélecteur CSS - caractère virgule (,)

- Permet de sélectionner plusieurs sélecteurs CSS en même temps
 - Parfait pour partager certaines déclarations CSS

```
/* Sélectionne tout élément ayant la classe "nouvelle" OU la classe "description" */  
.nouvelle,  
✓.description {  
  border:  red 2px solid;  
}
```

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/Selector_list

CSS – Sélecteur CSS - caractère "étoile" (*)

- Sélectionne toutes les balises
- **Ce sélecteur doit être utilisé avec extrême parcimonie**, des comportements inattendus peuvent apparaître

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/S%C3%A9lecteurs_de_classe

Pour aller plus loin et s'entraîner

jeu : <https://flukeout.github.io/> (en anglais)

Algorithme de résolution du CSS

CSS – Algorithme de résolution du CSS

Le navigateur fusionne tous les sélecteurs éligibles à un élément HTML lorsqu'il affiche une page mais...

CSS – Algorithme de résolution du CSS

...dans certains cas, il peut subsister des conflits au niveau des déclarations CSS.

Par exemple : deux sélecteurs appliquent la propriété "color", quelle valeur appliquer alors ?

CSS – Algorithme de résolution du CSS

Des règles, gérées par les navigateurs, ont été mises en place pour gérer ces cas. Qui sont représentés par les notions suivantes :

Source :

- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance

CSS – Algorithme de résolution du CSS

- Spécificité
 - Priorité des déclarations CSS d'un sélecteur
 - Peut être aussi appelé "poids du sélecteur"
- Cascade
 - Ordre des sélecteurs CSS dans le CSS
- Héritage
 - Gestion des propriétés CSS par rapport au parent

Source :

- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance

CSS – Spécificité

- Tous les sélecteurs CSS ne se valent pas
 - Certains ont un poids plus important
 - Leurs déclarations CSS seront prioritaires par rapport aux autres, et ce, quelque soit l'ordre du sélecteur dans le fichier

Sources :

- https://developer.mozilla.org/fr/docs/Web/CSS/Descendant_combinator
- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance#sp%C3%A9cificit%C3%A9

CSS – Spécificité

#mon-id > **.ma-classe**
.ma-classe > p

Les sélecteurs à gauche verront leurs déclarations CSS être appliquées par rapport à celles identique dans les sélecteurs à droite

Sources :

- <https://specificity.keegan.st/>
- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance#sp%C3%A9cificit%C3%A9

CSS – Spécificité

```
#fleur {  
  color: green;  
}
```

```
.fleur {  
  color: red;  
  margin: 1rem;  
}
```



```
{  
  color: green;  
  margin: 1rem;  
}
```

Déclaration appliquée au final sur l'élément (généré par le navigateur)

La spécificité donne l'avantage à la déclaration CSS "color: green;" car le sélecteur "#fleur" a un poids plus important. Toutefois si un élément possède la classe ".fleur" ET l'id "#fleur" alors la propriété "margin" s'appliquera aussi

CSS – Spécificité

#mon-id .enfant > **.ma-classe**
.parent .enfant > **.enfant**

Les sélecteurs à gauche verront leurs déclarations CSS être appliquées par rapport à celles identique dans les sélecteurs à droite.

On remarquera la présence d'espace entre les sélecteurs, ceci s'appelle le **sélecteur descendant**.

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/CSS/Descendant_combinator
- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance#sp%C3%A9cificit%C3%A9
- <https://melaniemeb.github.io/blog/2019/12/29/css-descendants-selectors.html>

CSS – Spécificité - Sélecteur descendants

- Il est possible d'écrire des sélecteurs plus précis en combinant plusieurs sélecteurs
- **Seul le dernier sélecteur de la liste voit les déclarations CSS être appliquées**

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/CSS/Descendant_combinator
- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance#sp%C3%A9cificit%C3%A9
- <https://melaniemeb.github.io/blog/2019/12/29/css-descendants-selectors.html>

CSS – Spécificité - Sélecteur descendants

- Combinaison représentée par un espace (minimum) entre chaque sélecteur
 - Chaque espace représente un niveau d'imbrication dans le HTML

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/CSS/Descendant_combinator
- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance#sp%C3%A9cificit%C3%A9
- <https://melaniemeb.github.io/blog/2019/12/29/css-descendants-selectors.html>

CSS – Spécificité - Sélecteur descendants

```
.herbier .fleur {  
  color: green;  
}
```

Élément qui
verra les
déclarations être
appliquées

Espace entre les
deux

Le style
s'applique
ici

```
<section class="herbier">
```

```
<p class="fleur">
```

Texte vert

```
</p>
```

```
</section>
```

CSS – Spécificité - Sélecteur descendants

```
.infos p {  
  color: blue;  
}
```



```
<article class="infos">  
  <p>Mon texte</p>  
</article>
```

```
.infos {  
  color: red;  
}
```

Le sélecteur ".infos p" est plus précis que ".infos" même si ce dernier est plus bas dans le code, le texte est donc bleu

CSS – Sélecteur descendants

- Attention à la sur-spécificité, les sélecteurs trop complexes rendent difficiles la lecture et la réutilisation du sélecteur

```
article section div.imbrication .element div p.description .text-pistil {  
  text-decoration: underline;  
  font-size: 25px;  
}
```



Très mauvaise pratique

CSS – Spécificité

- Les déclarations CSS directement dans l'attribut "style" possède la spécificité la plus élevée

CSS – Spécificité - !important

- Utilisation de la valeur spéciale "!important"
 - **A utiliser avec extrême parcimonie**
 - S'applique après la valeur d'une propriété CSS :
Exemple : `padding-left: 20px!important;`
 - Et lui permet de passer devant toutes les autres règles
 - Se situe en haut de toutes les règles de la spécificité du CSS
 - **A utiliser qu'en dernier cours**

Sources :

- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#Sp%C3%A9cificit%C3%A9
- <https://www.zonecss.fr/cours-css/poids-des-selecteurs-css.html>
- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#!important
- <https://specificity.keegan.st/> - calculateur de spécificité

CSS – Spécificité - !important

- Utilisé généralement pour outrepasser les **déclarations CSS** de fichiers CSS externes
 - Il vous arrivera de travailler avec des fichiers CSS externes que vous ne pourrez pas modifier

Sources :

- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance#sp%C3%A9cificit%C3%A9
- <https://www.zonecss.fr/cours-css/poids-des-selecteurs-css.html>
- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#!important
- <https://specificity.keegan.st/> - calculateur de spécificité

CSS – Spécificité

- Ordre de spécificité des **sélecteurs CSS** (du plus imposant au moins imposant) :
 1. Sélecteurs d'ID
 2. Sélecteurs de classe, d'attributs et pseudo-classes
 3. Sélecteurs de type (ou de balise) et pseudo-éléments

Pratiquons ensemble ! - Spécificité

<https://specificity.keegan.st/>

testeur de spécificité de sélecteurs CSS

Que se passe-t-il quand deux (ou plus) sélecteurs CSS ont la même spécificité ?

CSS – Cascade

- Fichier CSS est lu de haut en bas
 - Comme le HTML
- La déclaration la plus basse dans le fichier est prise en compte
 - **Si et seulement si la spécificité est identique**
 - Les sélecteurs CSS applicables sur un élément HTML sont fusionnés en prenant en compte les notions de spécificité et de cascade

Source :

- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance#la_cascade

CSS – Cascade

```
.paragraphe {  
  color: red;  
  text-decoration: underline;  
}
```


```
.paragraphe {  
  color: purple;  
}
```

Les balises <p> s'afficheront en violet car la déclaration concernant la couleur est la plus basse dans le fichier et elles seront soulignées car les déclarations sont fusionnées

Source :

- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance#la_cascade

CSS – Cascade

```
✓ .paragraphe {  
    text-decoration: underline;  
    color:  purple;  
}
```

Au final, c'est ce sélecteur qui sera appliqué (le tout a été fusionné par le navigateur)
(Ce sélecteur et son contenu sont générés par le navigateur)

Source :

- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance#la_cascade

CSS – Héritage

- Certaines propriétés CSS sont héréditaires :
 - Si un sélecteur a une déclaration, sa descendance en héritera sauf si elle surcharge la déclaration

Source :

- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance#h%C3%A9ritage

CSS – Héritage

- On peut forcer l'hérédité avec la valeur "inherit" pour la propriété que l'on souhaite

```
.text-pistil {  
  color: inherit;  
}
```

Ici on indique que la propriété "color" de la classe "text-pistil" va hériter de la valeur de son parent possédant une valeur pour cette propriété si elle n'est pas héritée par défaut

Source :

- https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance#h%C3%A9ritage

CSS – Héritage

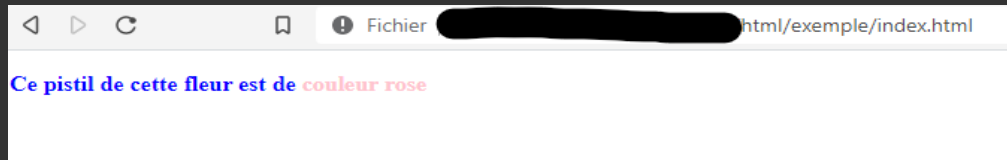
```
<body>
  <div>
    <p class="description">
      Ce pistil de cette fleur est de
      <span class="text-pistil"> couleur rose </span>
    </p>
  </div>
</body>
```

index.html

```
.text-pistil {
  color: pink;
}

.description {
  color: blue;
  font-size: 16px;
  font-weight: bold;
}
```

style.css



rendu dans le navigateur

Le texte de ma balise `` est en gras car il est contenu dans une balise `<p>` possédant la déclaration css "font-weight:bold;"

CSS – Héritage

| | |
|--|--|
| Applicabilité | tous les éléments. S'applique aussi à ::first-letter et ::first-line . |
| Héritée | oui |
| Valeur calculée | si la valeur est translucide, la valeur calculée est la fonction <code>rgba()</code> correspondante. Sinon, la fonction <code>rgb()</code> correspondante. Le mot-clé <code>transparent</code> devient <code>rgb(0,0,0)</code> . |

| | |
|--------------------------------|---|
| Applicabilité | tous les éléments exceptés ceux dont les types display pour les tableaux ne sont pas <code>table-caption</code> , <code>table</code> et <code>inline-table</code> . S'applique aussi à ::first-letter et ::first-line . |
| Héritée | non |

mdn vous indique si une propriété CSS est héritée ou non dans la partie "Spécifications" de la documentation d'une propriété CSS

Bonnes pratiques – Cascade, héritage et spécificité

- Privilégiez les sélecteurs de classe pour éviter les prises de tête
- Evitez les sélecteurs trop précis
 - Pensez à rendre vos classes réutilisables

CSS – Bonnes pratiques

- Éviter les caractères spéciaux dans les noms de sélecteurs :
 - Le tiret (-) et l'underscore (_) sont autorisés
- Mettre son CSS dans un fichier externe

CSS – Bonnes pratiques

- Nommer de façon claire et concise ses sélecteurs CSS
- Séparer le CSS en plusieurs fichiers si le fichier est trop conséquent
 - Attention à l'ordre de liage des fichiers CSS

CSS – Bonnes pratiques

- Éviter l'attribut id comme sélecteur css
- Éviter la redondance des règles
- Éviter les sélecteurs trop spécialisés
- Éviter les accents et autres caractères spéciaux (sauf '-' et '_')

CSS – Bonnes pratiques

- Éviter les sélecteurs directement sur la balise
 - Privilégier les classes
- Limiter le plus possible le CSS dans les balises HTML
- Éviter les noms de classes écrits en majuscules

CSS – Liens utiles pour trouver les propriétés de mise en forme des CSS

- MDN:
https://developer.mozilla.org/fr/docs/Learn/CSS/Styling_text
- W3C Schools :
<https://www.w3schools.com/cssref/>

CSS – Obtenir la définition d'une propriété sur VSCode

```
tre {
```

```
font-size: 12rem;
```

indicates the desired height of glyphs from the font. For scalable fonts, the font-size is a scale factor applied to the EM unit of the font. (Note that certain glyphs may bleed outside their EM box.) For non-scalable fonts, the font-size is converted into absolute units and matched against the declared font-size of the font, using the same absolute coordinate space for both of the matched values.

Syntax: <absolute-size> | <relative-size> | <length-percentage>

[MDN Reference](#)

Au clic on
atteint la
documentation

En survolant une propriété CSS, la popup affichée permet d'accéder directement à la doc MDN via un lien (MDN Reference)

Questions ?