

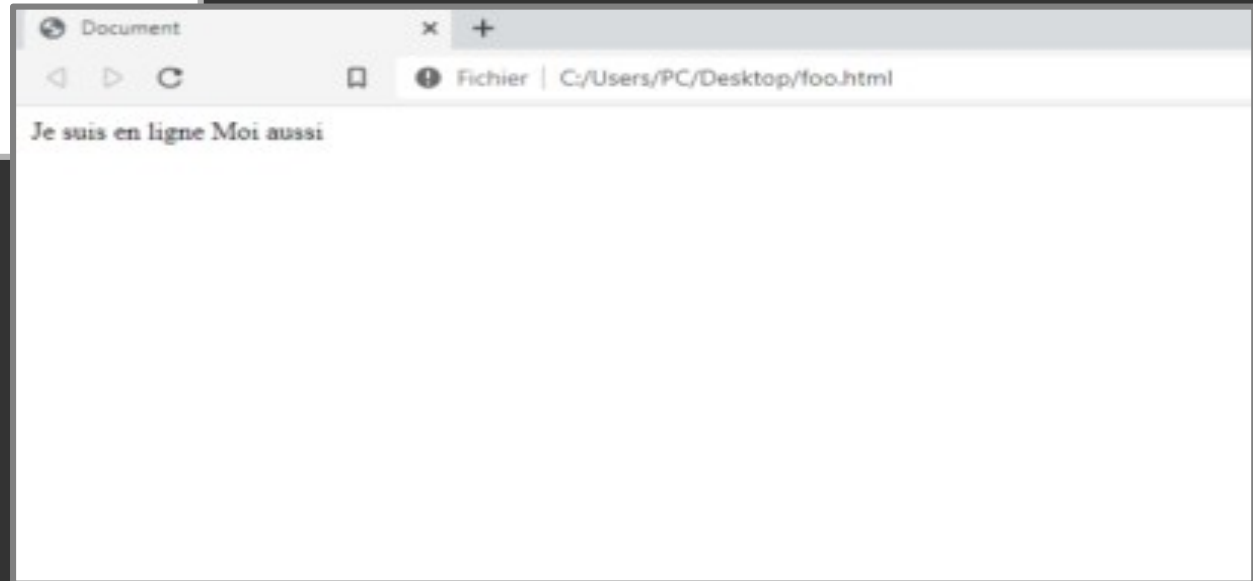
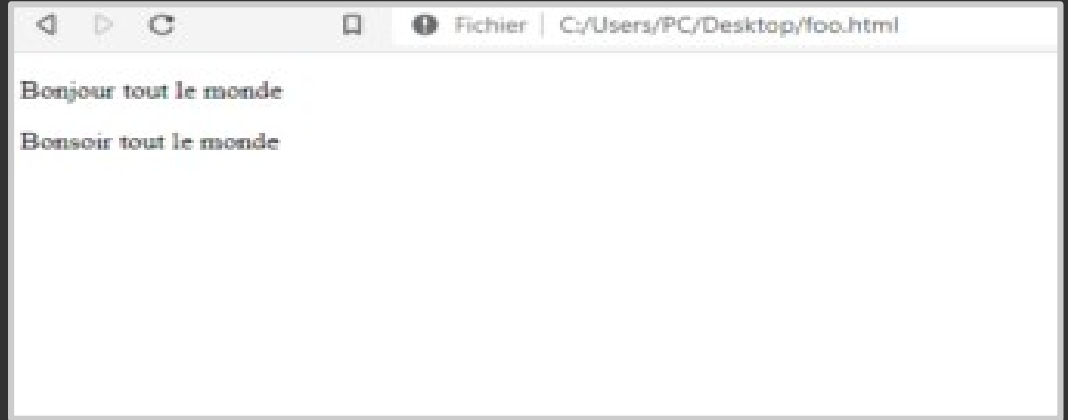
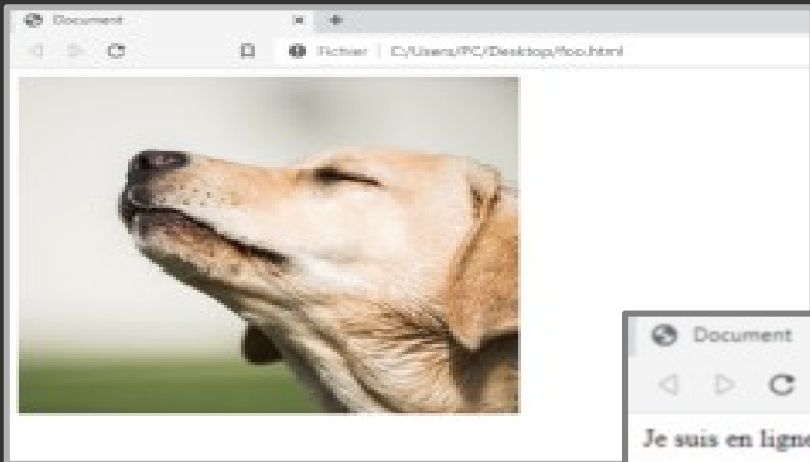
Intégration Web

MMI 1 – TP#2 S1



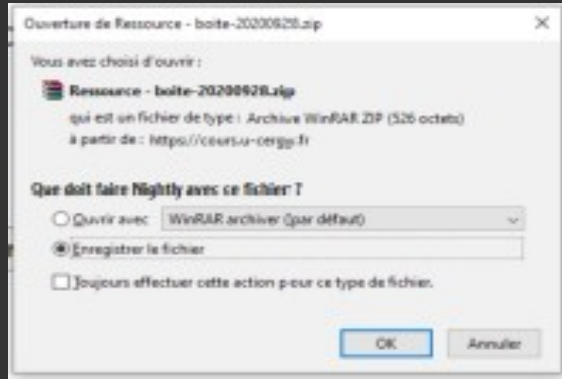
Danielo **JEAN-LOUIS**
Développeur front-end

Voilà ce que nous avons produit jusqu'à présent

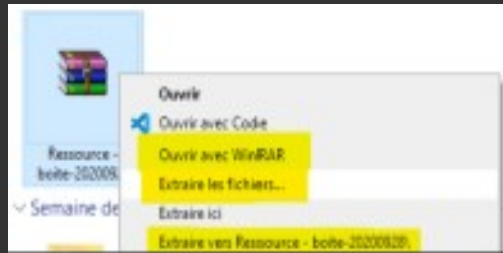


Pas très beau non ?

Petit point sur les ressources de cours

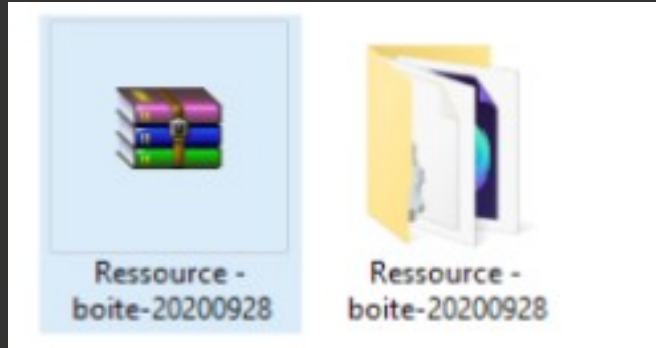


- Les ressources se présentent sous la forme d'une archive une fois téléchargées
- L'archive doit être impérativement extraite sinon vous ne serez pas capables de réaliser correctement les tp



Menu contextuel sur l'archive (clic droit). Choisir un des choix surligné.

Petit point sur les ressources de cours



- Une fois extrait, vous devez obtenir un dossier avec le contenu de l'archive
- Libre à vous de travailler dans le dossier OU en copier le contenu pour le coller ailleurs
- **En aucun vous devez travailler dans l'archive**
- Une fois fait, glissez le dossier dans VS Code

CSS – Feuilles de style

- Signifie Cascading Style Sheets
 - Feuilles de Style en Cascade
- Décrit la présentation d'une page html
- Version actuelle 3 (CSS3)
- Le langage évolue assez régulièrement

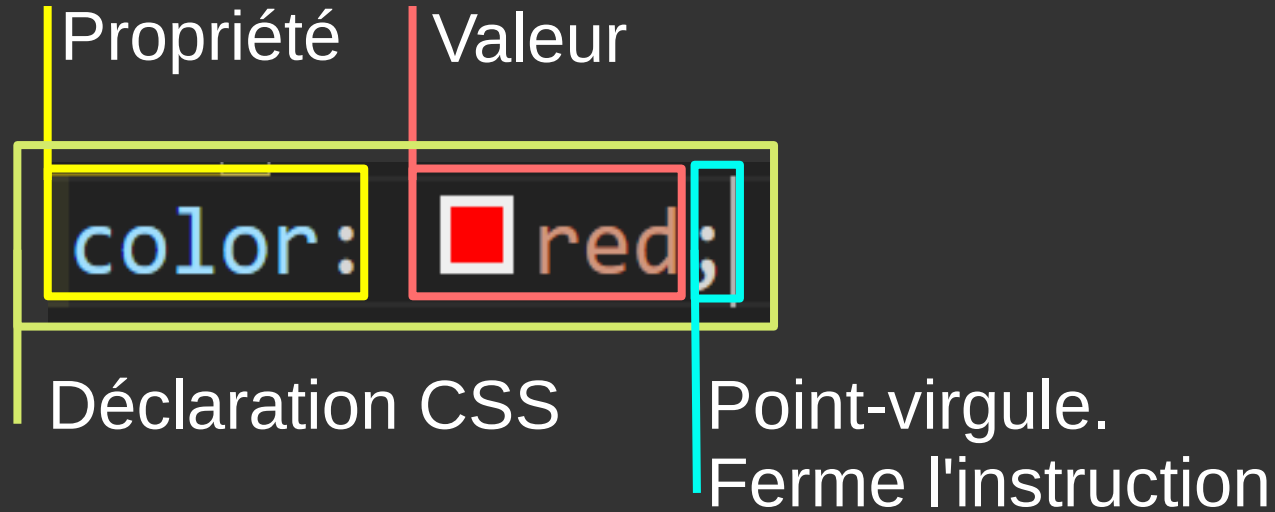
CSS – Feuilles de style

- Syntaxe :
 - Sélecteur : élément sur lequel le style s'applique
 - Propriété : effet que l'on souhaite donner (taille, couleur de texte, espacement)
 - Valeur : valeur de la propriété ("pink", 42px, etc.)

CSS – Feuilles de style

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

CSS – Feuilles de style - syntaxe



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



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

- La balise html fait office de sélecteur

Pratiquons ! - Écrivons notre premier CSS

Pré-requis :

- Avoir la ressource `ressources/css-balises/`

A télécharger ici :

<https://download-directory.github.io/?url=https://github.com/DanYellow/cours/tree/main/integration-web-s1/travaux-pratiques/numero-2/ressources>

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+ valeurs 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 possible sur la même page de la balise
- Utilise un système de **sélecteur**

Source :

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

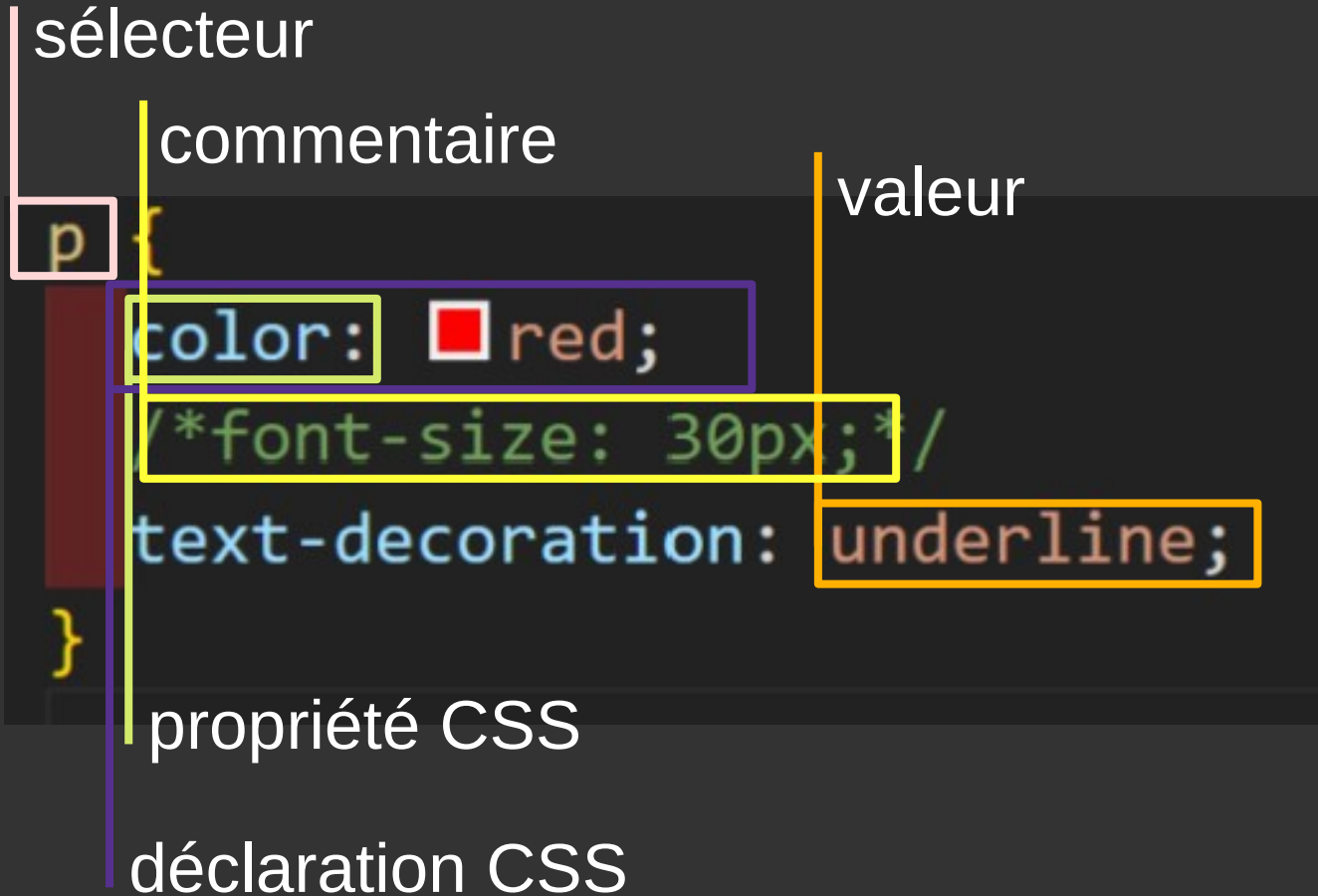
CSS – Sélecteur css

- 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 et donc ses déclarations 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



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 ! - Première balise <style>

Pré-requis :

- Avoir la ressource [ressources/css-style/](#)

A télécharger ici :

<https://download-directory.github.io/?url=https://github.com/DanYellow/cours/tree/main/integration-web-s1/travaux-pratiques/numero-2/ressources>

CSS – Balise <style> - Avantage

- Séparation de la logique
 - La définition du style est séparée du 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 :

- <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

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 de classe

- Contrevient à limitation des sélecteurs de balises
 - 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 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 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

Source :

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

CSS – Sélecteur de classe

- Une balise HTML peut avoir plusieurs classes CSS (séparée 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 de classe - syntaxe

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

Sélecteur ici
de classe

Source :

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

CSS – Sélecteur de classe - syntaxe

A diagram illustrating the syntax of a CSS class selector. It shows the text ".consignes-conteneur" in a monospace font. A red square highlights the period (dot) character. A red line extends from the bottom of this square down to the text "un point". A red bracket extends from the right side of the text ".consignes-conteneur" down to the text "nom de classe".

`.consignes-conteneur`

un point

nom de classe

- Sélecteur de classe : "." + "nom-de-classe"
- Sélecteur d'id : "#" + "nom-de-id"

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 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 de classe

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

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



Le style ne s'applique pas

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



Le style s'applique

CSS – Sélecteur de classe

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

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 ! - Premier fichier .css

Pré-requis :

- Avoir la ressource `ressources/css-link/`

A télécharger ici :

<https://download-directory.github.io/?url=https://github.com/DanYellow/cours/tree/main/integration-web-s1/travaux-pratiques/numero-2/ressources>

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>
```

- L'ordre des classes dans l'attribut "class" n'influe en rien sur l'interprétation des déclarations
 - C'est dans le fichier .css (<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

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

Source :

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

CSS – Sélecteur CSS

- Imbrication représentée par un espace entre chaque sélecteur
- Appelé "Sélecteurs descendant"

Source :

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

CSS – Sélecteur CSS - Imbrication

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

Espace entre les
deux

Élément qui
verra les
déclarations être
appliquées

Le style
s'applique
ici

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

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

Texte vert

```
</p>
```

```
</section>
```

CSS – Sélecteur de classe

- L'attribut "id" peut être également utilisé pour le même rôle, mais on privilégiera l'attribut "class". Les id sont ciblés par un croisillon "#" dans le css
- Le point précédent s'applique également aux attributs

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
- 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

- 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" />
```

Le style ne s'applique qu'aux imprimantes grâce à l'attribut "media" et la valeur "print"

Pratiquons ! - Premier fichier .css

Pré-requis :

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

A télécharger ici :

<https://download-directory.github.io/?url=https://github.com/DanYellow/cours/tree/main/integration-web-s1/travaux-pratiques/numero-2/ressources>

CSS – Sélecteur de balises "*"

- 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)

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
- Nommer de façon claire et concise ses sélecteurs CSS
- Séparer le css en plusieurs fichiers si le fichier est trop conséquent

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

Cascade, Héritage

Cascade et Héritage

- Notions clés du CSS
 - Complément de la spécificité CSS
 - Evite les prises de tête
- Cascade
 - Gestion de l'ordre des sélecteurs dans le fichier
- Héritage
 - Gestion des propriétés CSS à par rapport au parent

Sources :

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

CSS – Feuilles de style – La cascade

- Fichier CSS est lu de haut en bas
 - Comme le HTML
- La déclaration la plus basse est prise en compte
 - Si la spécificité est identique

Source :

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

CSS – Feuilles de style – La cascade

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

```
p {  
  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

Source :

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

CSS – Feuilles de style – Héritage

- Certaines propriétés CSS sont héréditaires :
 - Si une classe / balise html 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 – Feuilles de style – 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, le cas échéant, elle prendra la valeur par défaut.

Source :

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

CSS – Feuilles de style – Héritage

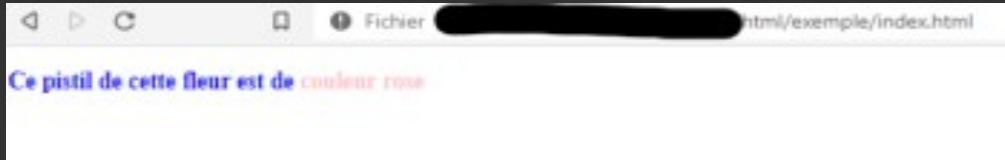
```
<body>
  <div>
    <p class="description">
      Ce pistil de cette fleur est de
      <span class="text-pistil"> couleur rose </span>
    </p>
  </div>
</body>
```

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 – Feuilles de style – Imbrication

- Tout comme les balises html, les sélecteurs css peuvent s'imbriquer

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

Ici je sélectionne toute classe "text-pistil" contenue dans une balise <p> ayant la classe "description" elle-même contenue dans une balise <div>

CSS – Feuilles de style – Imbrication

- Attention à la sur-imbrication, 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;  
}
```



Mauvaise pratique

Pratiquons ! - Cascade et héritage

Pré-requis :

- Avoir la ressource `ressources/cascade-et-heritage/`

A télécharger ici :

<https://download-directory.github.io/?url=https://github.com/DanYellow/cours/tree/main/integration-web-s1/travaux-pratiques/numero-2/ressources>

CSS – Récapitulatif

- Le CSS est un langage permettant d'ajouter du style aux pages html
- Externalisé le CSS s'écrit dans un fichier .css
- Les propriétés CSS s'appliquent en cascade
- Certaines propriétés CSS sont héritées

Questions ?