

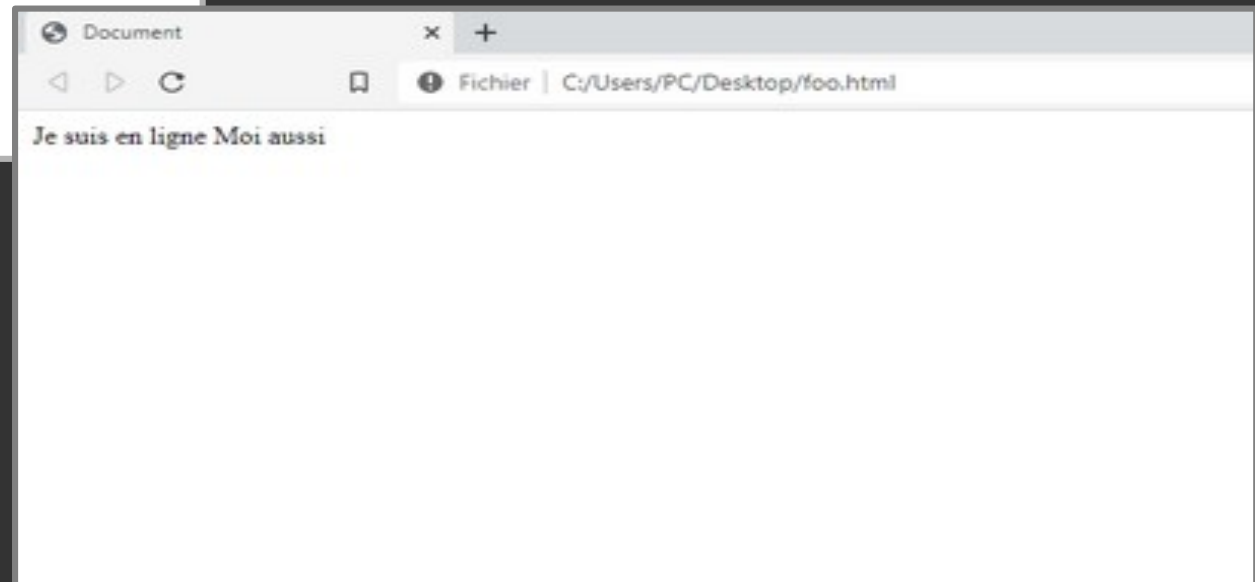
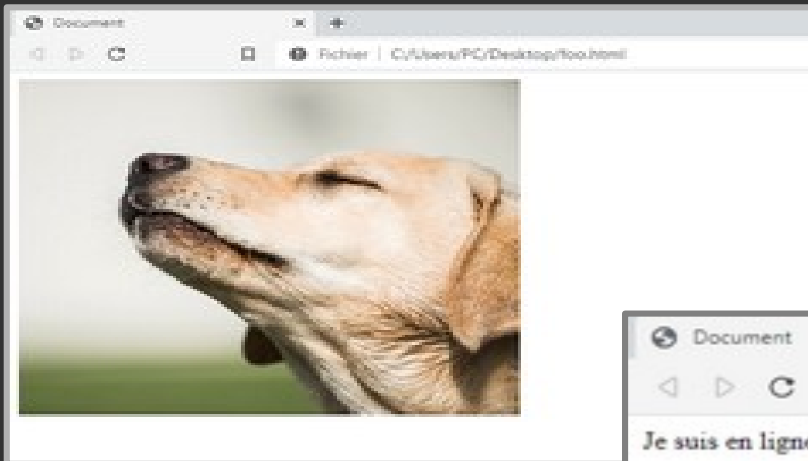
Intégration Web

MMI 1 – TP#2 S1



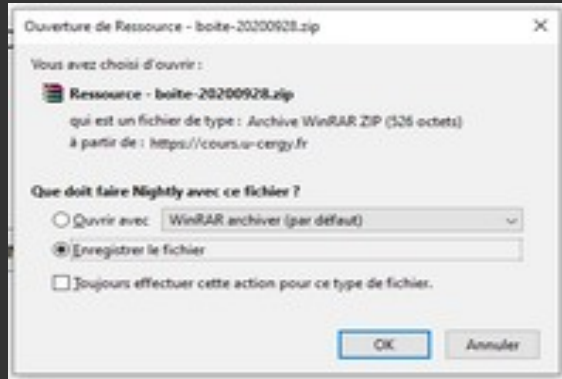
Danielo **JEAN-LOUIS**
Développeur front-end

Voilà ce que nous avons produit jusqu'à présent

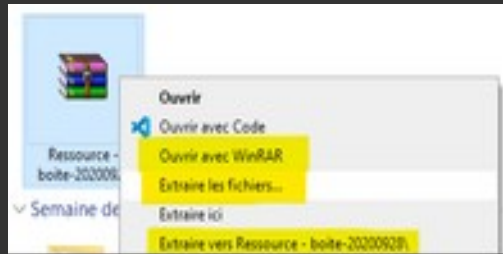


Pas très beau non ?

Petit point sur les ressources de cours

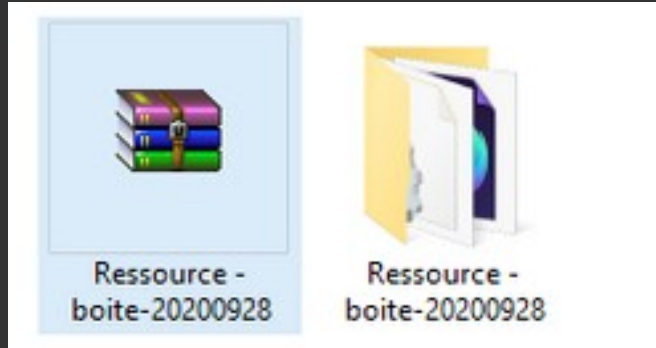


- Les ressources se présentent sous la forme d'une archive une fois téléchargées
- L'archive doit être impérativement extraite sinon vous ne serez pas capables de réaliser correctement les tp



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

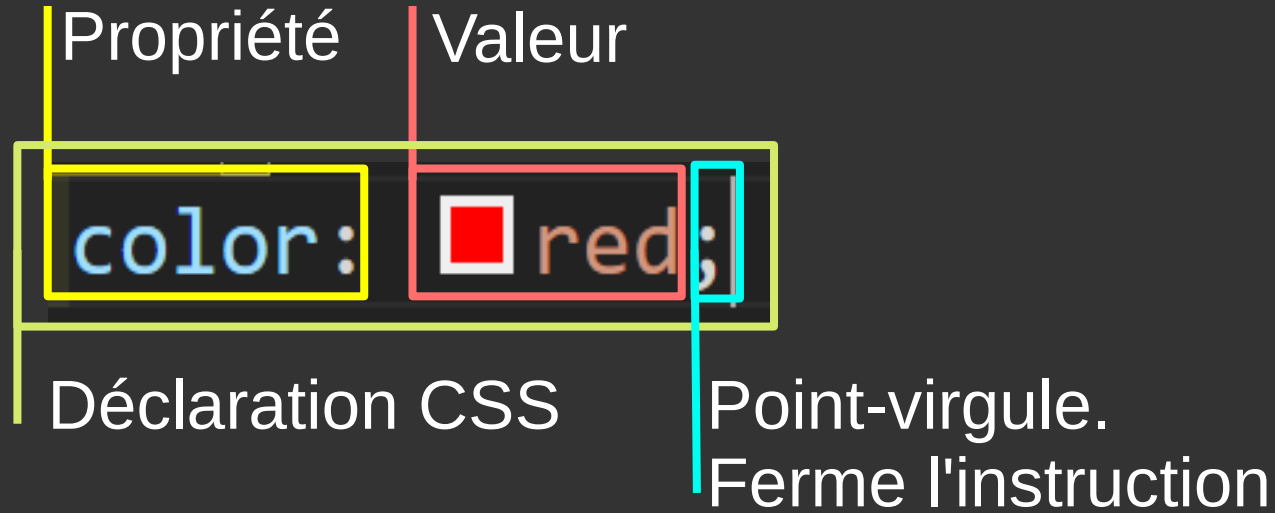
CSS – Feuilles de style

- 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



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

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>

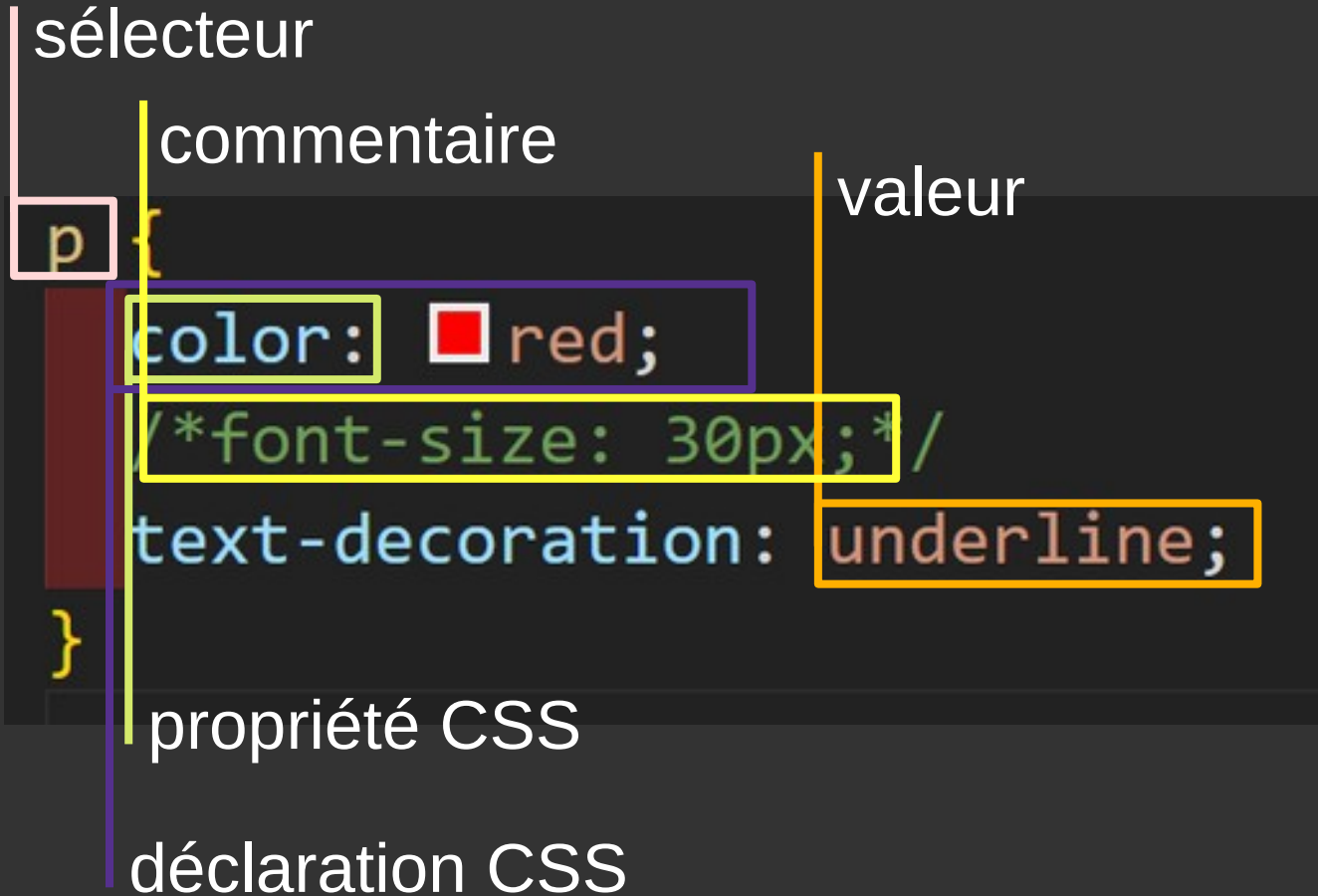
CSS – Sélecteur css

- 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> - Inconvénient

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

Mauvaise pratique – sélecteur de balises

- Crée des effets de bord
 - Affecte tous les éléments ayant la même balise
 - Rend l'écriture de sélecteur laborieuse

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

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 sélecteur

Source :

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

CSS – Sélecteur de classe

- Contrevient à limitation des sélecteurs html
 - Un sélecteur html 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 classe peut être utilisée par plusieurs balises html
 - Même si elles sont différentes
 - Réutilisable
- Un nom de classe est arbitraire MAIS il est préférable qu'il soit explicite
 - Éviter les accents et autres caractères spéciaux (sauf '-' et '_')

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

- Lié aux balises grâce à l'attribut "class"
- Les noms de classes sont sensibles à la casse :
 - **.MaClasse** ce n'est pas la même chose que **".maclasse"**

Source :

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

CSS – Sélecteur de classe

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

sélecteur de
classe

nom de
classe

Source :

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

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="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**

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

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 – Feuilles de style – La cascade

- Comme le html, le navigateur lit un fichier css de haut en bas. Si une déclaration est redéfinie plus tard, c'est celle la plus basse qui est prise en compte

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

Les balises <p> s'afficheront en violet car la déclaration concernant la couleur est la plus basse dans le fichier

```
p {  
  color: purple;  
}
```

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

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

Source :

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

CSS – Feuilles de style – La cascade

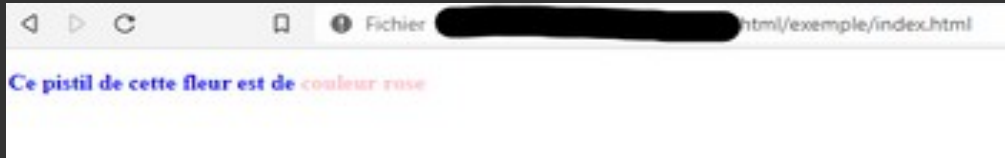
```
<body>
  <div>
    <p class="description">
      Ce pistil de cette fleur est de
      <span class="text-pistil"> couleur rose </span>
    </p>
  </div>
</body>
```

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

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

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 – 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 ?