

# Intégration Web

MMI 1 – TP#4 S1



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

# Formatage du texte - Généralités

- Fonctionne sur toutes les formes de texte
  - Balises : `<a>`, `<p>`, `<span>`, `<li>`...
- Permet de structurer, hiérarchiser l'information
  - Rend l'information plus simple à trouver

Source(s) :

- [https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction\\_to\\_HTML/HTML\\_text\\_fundamentals](https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction_to_HTML/HTML_text_fundamentals)

# Hiérarchie du texte - Généralités

- Fonctionne via le CSS pour les internautes voyants
- Fonctionne via le HTML (sémantique) pour les internautes non-voyants

## Source(s) :

- [https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction\\_to\\_HTML/HTML\\_text\\_fundamentals](https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction_to_HTML/HTML_text_fundamentals)

# Formatage du texte

- Exclusivement fait grâce aux propriétés CSS
  - color : couleur du texte
  - font-weight : graisse du texte
  - font-family : police du texte
  - font-size : taille du texte
  - text-align : alignement du texte
  - text-decoration : "décore" le texte
  - text-transform : change la casse du texte
  - [...]

## Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/CSS/CSS\\_Text](https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Text)
- <https://developer.mozilla.org/fr/docs/Web/CSS/text-decoration>
- [https://developer.mozilla.org/fr/docs/Learn/CSS/Styling\\_text/initiation-mise-en-forme-du-texte](https://developer.mozilla.org/fr/docs/Learn/CSS/Styling_text/initiation-mise-en-forme-du-texte)
- <https://cssgradient.io/blog/css-gradient-text/> - texte en dégradé - en

## Formatage du texte

**N'utilisez pas des balises comme  
<center> pour formater le texte.  
Elles sont obsolètes.**

# Point accessibilité – Casse du texte

- Évitez les paragraphes écrits en majuscules, ceci rend la lecture compliquées pour les personnes dyslexiques notamment
  - Limiter les majuscules aux titres ou textes courts

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/text-transform#accessibilit%C3%A9>

# Formatage du texte - font-weight

- Accepte comme valeur :
  - normal (400) (valeur par défaut) / bold (700)
  - Nombres compris en 100 et 900 définis
    - Dépendra de la police
  - ...

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/font-weight>



# Formatage du texte - font-family

- Permet de changer la police du texte
- Accepte comme valeur une liste de polices ordonnée par priorité, la police prioritaire étant la plus à gauche
  - Exemple : **Cambria**, Cochin, Georgia, Times, 'Times New Roman', **serif**;

## Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/font-family>
- <https://www.cssfontstack.com/> - liste des polices "sûres"
- <https://developer.mozilla.org/fr/docs/Web/CSS/@font-face>

# Formatage du texte - font-family

```
.ma-classe {  
  font-family: Cambria, 'Times New Roman', serif;  
}
```

Si la police "Cambria" n'est pas trouvée alors on charge "Times New Roman", etc.

## Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/font-family>
- <https://www.cssfontstack.com/> - liste des polices "sûres"
- <https://developer.mozilla.org/fr/docs/Web/CSS/@font-face>

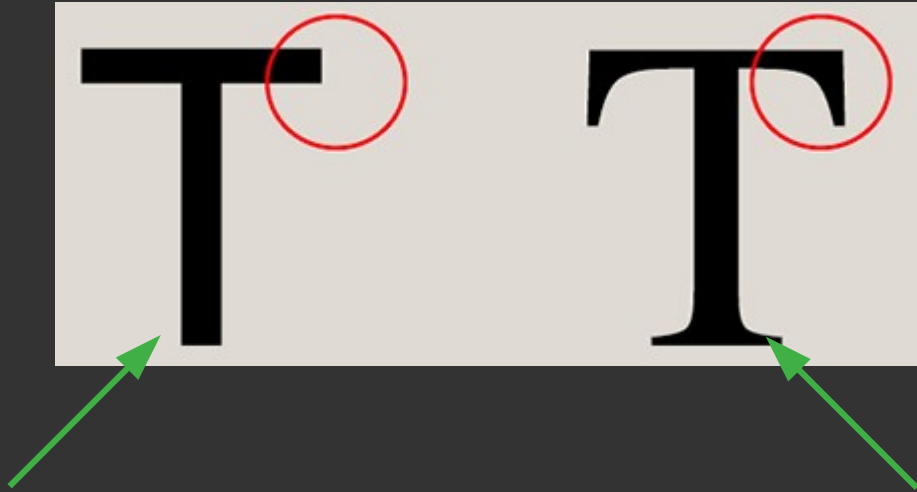
# Formatage du texte – serif ou sans-serif

- Valeurs de secours si celles précédentes ne fonctionnent pas
  - On met l'un **OU** l'autre à la fin de la liste de polices
- Indispensable pour éviter d'avoir de mauvaises surprises

## Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/font-family>
- <https://www.cssfontstack.com/> - liste des polices "sûres"
- <https://developer.mozilla.org/fr/docs/Web/CSS/@font-face>

# Formatage du texte – serif ou sans-serif



## **Sans-serif (sans empattement)**

- Représente la modernité
- Exemple de police : Helvetica, Futura, Arial...
- Plus présent sur le web

## **Serif (avec empattement)**

- Représente le sérieux, le professionnalisme
- Arrivé à l'époque de Gutenberg
- Exemple de polices : "Times New Roman", Georgia ...

## **Source(s) :**

- <https://www.adobe.com/fr/creativecloud/design/discover/serif-vs-sans-serif.html>
- <https://www.get-beyond.fr/serif-sans-serif-difference/>
- <https://www.canva.com/learn/serif-vs-sans-serif-fonts/> - anglais

# Formatage du texte - font-family

- Par défaut, accès uniquement à des polices système
- Utilisation de la règle @font-face pour utiliser des polices non système
  - On ne l'utilisera pas dans le cadre du cours

## Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/font-family>
- <https://www.cssfontstack.com/> - liste des polices "sûres"
- <https://developer.mozilla.org/fr/docs/Web/CSS/@font-face>

# Formatage du texte - font-family

- Listez des polices cohérentes dans votre propriété
  - Ne mettez pas "Comic Sans MS" et "Impact" ensemble, par exemple
  - Note : VS Code propose des polices qui vont ensemble à l'auto-complétion

## Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/font-family>
- <https://www.cssfontstack.com/> - liste des polices "sûres"
- <https://developer.mozilla.org/fr/docs/Web/CSS/@font-face>

# Formatage du texte - font-family



**Abc**

Impact

Abc

Comic Sans MS

Abc

Arial

Les trois polices sont à 40px  
pourtant elles n'occupent pas le  
même espace d'où l'importance  
d'avoir une liste de polices  
cohérentes pour la valeur de la  
propriété "font-family"

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/font-family>
- <https://www.cssfontstack.com/> - liste des polices "sûres"
- <https://developer.mozilla.org/fr/docs/Web/CSS/@font-face>

# Variables / Propriétés personnalisées CSS

- Éviter la redondance des **valeurs de propriétés**
- Géré par tous les navigateurs modernes

Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/CSS/Using\\_CSS\\_custom\\_properties](https://developer.mozilla.org/fr/docs/Web/CSS/Using_CSS_custom_properties)



# Variables / Propriétés personnalisées CSS

```
.mon-selecteur {  
    /* Définition de la variable */  
    --couleur-marque: rgb(14, 25, 90);  
  
    /* utilisation de la variable */  
    color: var(--couleur-marque, blue);  
}
```

fonction "var" pour pouvoir utiliser  
les propriétés personnalisées

propriété  
personnalisée  
définie avant

valeur par défaut si  
la propriété n'existe  
pas (facultatif)

une variable CSS

Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/CSS/Using\\_CSS\\_custom\\_properties](https://developer.mozilla.org/fr/docs/Web/CSS/Using_CSS_custom_properties)

# Variables / Propriétés personnalisées CSS

- Les variables possèdent une portée (scope) définie par le sélecteur CSS
- Utilisation de la pseudo-classe ":root" pour rendre la variable globale

```
:root {  
  --couleur-arriere-plan: blue;  
}  
  
body {  
  --couleur-verte: rgb(0, 201, 50);  
  background-color: var(--couleur-arriere-plan, blue);  
  color: var(--couleur-verte);  
  border: 3px solid var(--couleur-verte);  
}
```

La variable est globale car définie dans ":root"

Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/CSS/Using\\_CSS\\_custom\\_properties](https://developer.mozilla.org/fr/docs/Web/CSS/Using_CSS_custom_properties)

# Formatage du texte - font-family

```
:root {  
  --liste-polices: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;  
}
```

```
.article-contenu {  
  font-family: var(--liste-polices);  
  /* [...] */  
}
```

```
.article-titre {  
  font-family: var(--liste-polices);  
  /* [...] */  
}
```

```
.article-contenu {  
  font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;  
  /* [...] */  
}
```

```
.article-titre {  
  font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;  
  /* [...] */  
}
```

Les deux codes font la même chose mais l'utilisation de variables rend le code plus facile à modifier

**Vous n'êtes pas obligé d'utiliser  
des variables, c'est pour votre  
confort de travail**

# Formatage du texte - Règles

- Préférez la balise `<p>` pour contenir un bloc de texte
- Pour formater du texte dans une balise `<p>`, il faut utiliser la balise `<span>`
- Une balise `<p>` ne peut pas être dans une autre balise `<p>`
  - **Il faut utiliser la balise `<span>`**

Source(s) :

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

# Formatage du texte - Règles





```
<p class="centre">
  Ceci <p class="gras">est interdit</p>
</p>
```

**Incorrect :**

Il y a une balise `<p>` dans une autre balise `<p>`

**Correct :**

Il y a une balise `<span>` pour styliser dans une balise `<p>`



```
<p class="centre">
  Ceci <span class="gras">est correct</span>
</p>
```

Source(s) :

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

# Formatage du texte - Règles

- Il est possible d'imbriquer des balises `<span>`

## Source(s) :

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

## font-family – Bonnes pratiques

- Évitez de mélanger trop de polices d'écriture au sein même d'un site  
**Deux ou trois** polices différentes maximum
- Pensez aux dyslexiques, préférez les polices simples à celles (trop) stylisées

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/font-family>



# Formatage du texte - font-size

- Définit la taille du texte
- Accepte comme valeurs (toujours positives) :
  - Pourcentage (%)
  - Longueur, unités possibles :
    - px
    - **rem**
    - em
    - ...

## Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/font-size>
- <https://developer.mozilla.org/fr/docs/Web/CSS/length>

# Point accessibilité – Taille de texte

- 16 px est considéré comme étant la taille recommandée pour un corps de texte sur le web
  - En dessous ça devient compliqué à lire
- Préférer les tailles de polices relatives comme le rem, le % ou l'em. **Mais le rem reste moins prise de tête pour les calculs**
  - Ne pas oublier que certains utilisateurs peuvent zoomer ou même changer la taille de base de la font-size du navigateur

# Formatage du texte – rem / em / %

- Unités relatives
- rem : basé sur l'élément racine (r pour root) de la page (balise <html> ou navigateur) avec une valeur de 16px
- em / % : plus utilisé pour les marges notamment de la balise <p>
  - Se base sur la font-size du parent **direct**

**On préférera utiliser l'unité "rem"**

Source(s) :

- [https://developer.mozilla.org/fr/docs/Learn/Common\\_questions/Design\\_and\\_accessibility/Design\\_for\\_all\\_types\\_of\\_users#les\\_units%C3%A9s\\_relatives](https://developer.mozilla.org/fr/docs/Learn/Common_questions/Design_and_accessibility/Design_for_all_types_of_users#les_units%C3%A9s_relatives)

# Formatage du texte – rem

- 1 rem représente 100 % de la valeur de la font-size du root (racine)
- S'applique à toutes les propriétés CSS acceptant une longueur (width, font-size, gap, margin...)
- Indispensable avec la propriété "font-size" pour l'accessibilité

## Source(s) :

- [https://developer.mozilla.org/fr/docs/Learn/Common\\_questions/Design\\_and\\_accessibility/Design\\_for\\_all\\_types\\_of\\_users#les\\_units%C3%A9s\\_relatives](https://developer.mozilla.org/fr/docs/Learn/Common_questions/Design_and_accessibility/Design_for_all_types_of_users#les_units%C3%A9s_relatives)

**Utilisez toujours l'unité rem au moins  
pour la propriété « font-size »**

# Pratiquons ! - Exemple unité rem

Pré-requis :

- Avoir la ressource ressources/exemple-unite-rem

A télécharger ici :

[https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-4/integration-web-s1\\_travaux-pratiques\\_numero-4.ressources.zip](https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-4/integration-web-s1_travaux-pratiques_numero-4.ressources.zip)

## Formatage du texte – rem

- Photoshop (et autres) utilisent l'unité pixel
  - Il faudra donc faire la conversion px → rem
- Formule conversion pixel → rem :

$$\text{taille en rem} = \frac{\text{taille police \acute{e}l\acute{e}ment (px)}}{\text{taille police racine* (px)}}$$

\* : Taille de police par d\acute{e}faut du navigateur, cette valeur est souvent 16.

## Formatage du texte – em / %

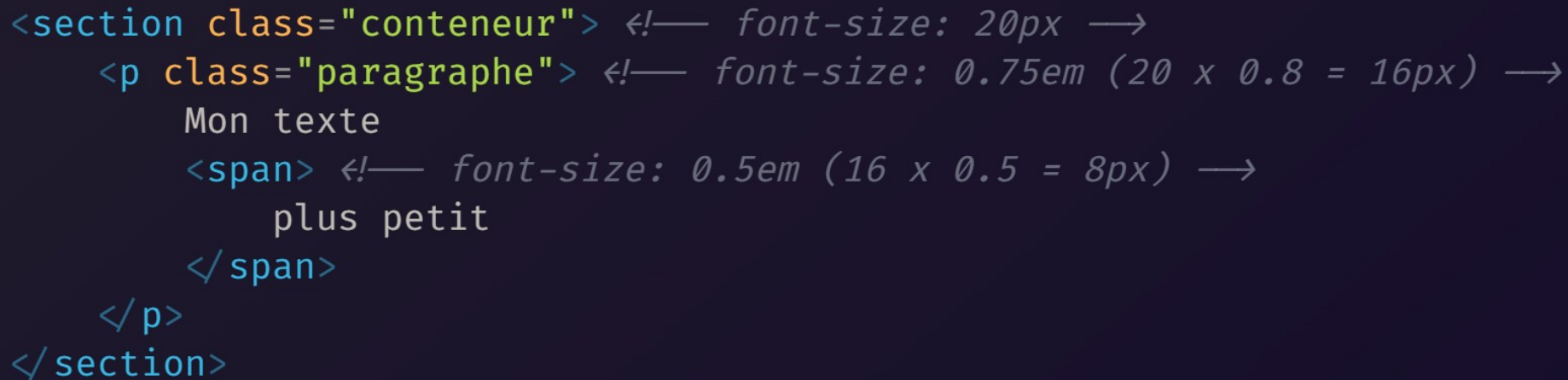
- 1 em / 100 % représente 100 % de la valeur de la **font-size du parent direct**
- Calcul pixel → em / % :

$$\text{taille en em/\%} = \frac{\text{taille police \acute{e}l\acute{e}ment (px)}}{\text{taille police parent (px)}}$$



# Formatage du texte – em / %

Exemple :



```
<section class="conteneur"> ←!— font-size: 20px →
  <p class="paragraphe"> ←!— font-size: 0.75em (20 x 0.8 = 16px) →
    Mon texte
    <span> ←!— font-size: 0.5em (16 x 0.5 = 8px) →
      plus petit
    </span>
  </p>
</section>
```

Ce raisonnement s'applique à "l'infini"

**Privilégiez l'unité rem pour vous  
éviter des prises de tête et être  
conforme aux règles  
d'accessibilité**

# Formatage du texte – text-align

- Permet de modifier l'alignement du texte
- Aligné à gauche par défaut (text-align: start;)

## Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/text-align>
- <https://developer.mozilla.org/fr/docs/Web/CSS/text-justify>

# Formatage du texte – text-align

- Accepte les valeurs suivantes :
  - start, end, center, justify
- La valeur "justify" donne accès à une nouvelle propriété "text-justify" pour ajouter de la granularité à la justification du texte

## Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/text-align>
- <https://developer.mozilla.org/fr/docs/Web/CSS/text-justify>

# Formatage du texte – text-align

- Historiquement, on utilisait
  - "left" pour aligner à gauche (maintenant "start")
  - "right" pour aligner à droite (maintenant "end")

## Source(s) :

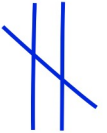
- <https://developer.mozilla.org/fr/docs/Web/CSS/text-align>
- <https://developer.mozilla.org/fr/docs/Web/CSS/text-justify>

**"start" ou "left", "end" ou "right", ils sont tous valides, mais préférez la version moderne ("start" et "end")**

# Point accessibilité – Alignement du texte

- La justification du texte dans le web est problématique
  - Crée des lézardes (voir ci-contre)
  - Rend difficile la lecture notamment pour les dyslexiques
- Évitez la valeur "justify" pour la propriété "text-align" sur le web

ecte turnadis pisici elit, sed  
modest empora incidunt ut  
re et dolore magma aliqua.  
enimenad mi meniat, quis  
rud exercitaton ullamco  
ris nisi utaliniq id ex ea  
modi consequat. Quis aute  
repreheni deriti voliptate  
resse cillum doloresu fugiat  
n pariatur. Excepteuren asint  
ecat cupiditat non proidens



Les lézardes rendent compliqué la lecture, l'œil a du mal à suivre

## Source(s) :

- <https://designforhackers.com/blog/justify-text-html-css> – anglais
- [www.codebuilder.fr/blog/redaction/justifier-texte-web-bonne-pratique/](http://www.codebuilder.fr/blog/redaction/justifier-texte-web-bonne-pratique/)
- [https://fr.wikipedia.org/wiki/L%C3%A9zarde\\_\(imprimerie\)](https://fr.wikipedia.org/wiki/L%C3%A9zarde_(imprimerie))

# Point accessibilité – Alignement du texte

- Le centrage du texte est aussi problématique
  - Trop de textes centrés rendent compliqué la lecture : l'utilisateur a du mal à "suivre" les lignes
  - Préférez le centrage uniquement pour des textes courts ou titres

## Sources :

- <https://designforhackers.com/blog/justify-text-html-css> – anglais
- [www.codebuilder.fr/blog/redaction/justifier-texte-web-bonne-pratique/](http://www.codebuilder.fr/blog/redaction/justifier-texte-web-bonne-pratique/)



# Formatage du texte – text-decoration

- Décore le texte
- Propriété raccourcie regroupant :
  - text-decoration-line
  - text-decoration-color
  - text-decoration-style
- Permet par exemple de souligner ou barrer du texte avec une certaine couleur

## Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/text-decoration>
- [https://developer.mozilla.org/fr/docs/Web/CSS/CSS\\_Text\\_Decoration](https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Text_Decoration)

## Point accessibilité – Soulignement

- Réservé historiquement pour les (hyper)liens (balise <a>)
  - **A utiliser avec parcimonie si ce n'est pas pour un lien**


# Formatage du texte – listes

- Permet de lister des éléments
  - `<ul>` : non ordonnée. Ex : liste de courses
  - `<ol>` : ordonnée. Ex : classement
- Chaque élément de liste doit être contenu dans une balise `<li>`

## Source(s) :

- [https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction\\_to\\_HTML/HTML\\_text\\_fundamentals#listes](https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction_to_HTML/HTML_text_fundamentals#listes)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/ul>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/li>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/ol>

# Formatage du texte – listes



```
<ul>
  Bonjour !
</ul>
```



**Incorrect :**  
Il y a du texte directement dans la balise `<ul>`

**Correct :**  
Il y a une balise `<li>` qui contient le texte



```
<ul>
  <li>Bonjour !</li>
</ul>
```



## Source(s) :

- [https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction\\_to\\_HTML/HTML\\_text\\_fundamentals#listes](https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction_to_HTML/HTML_text_fundamentals#listes)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/ul>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/li>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/ol>

# Formatage du texte – listes

- Une liste peut en contenir une autre si et seulement si la sous-liste est contenue dans une balise `<li>`
- Une liste `<ul>` peut contenir une sous-liste `<ol>` et vice-versa

## Source(s) :

- [https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction\\_to\\_HTML/HTML\\_text\\_fundamentals#listes](https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction_to_HTML/HTML_text_fundamentals#listes)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/ul>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/li>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/ol>

# Formatage du texte – listes

- Une balise `<li>` doit être dans une balise `<ul>` ou `<ol>`. **Nulle part ailleurs.**

## Source(s) :

- [https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction\\_to\\_HTML/HTML\\_text\\_fundamentals#listes](https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction_to_HTML/HTML_text_fundamentals#listes)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/ul>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/li>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/ol>

# Pratiquons ! - Formatage texte

Pré-requis :

- Avoir la ressource ressources/formatage-texte

A télécharger ici :

[https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-4/integration-web-s1\\_travaux-pratiques\\_numero-4.ressources.zip](https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-4/integration-web-s1_travaux-pratiques_numero-4.ressources.zip)

# Formatage du texte – Titres

- Balise `<h1>`-`<h6>`
  - Plus le chiffre est petit plus le titre est important
  - Permet de définir du texte important du point de vue référencement
  - Génère une table des matières

Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/HTML/Element/Heading\\_Elements](https://developer.mozilla.org/fr/docs/Web/HTML/Element/Heading_Elements)





# Formatage du texte – Titres

- Impossibilité de mettre plusieurs h1 dans la même page
- Interdiction de faire des sauts de niveau de titre

Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/HTML/Element/Heading\\_Elements](https://developer.mozilla.org/fr/docs/Web/HTML/Element/Heading_Elements)



# Formatage du texte – Titres



```
<h1>Titre de niveau 1</h1>  
<h3>Titre de niveau 3</h3>  
<h5>Titre de niveau 5</h5>
```

Ce code est invalide car il saute des niveaux de titre

Ce code est valide car il ne saute pas des niveaux de titre



```
<h1>Titre de niveau 1</h1>  
<h2>Titre de niveau 2</h2>  
<h3>Titre de niveau 3</h3>
```

Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/HTML/Element/Heading\\_Elements](https://developer.mozilla.org/fr/docs/Web/HTML/Element/Heading_Elements)

# Formatage du texte – Titres

- Si vous ne savez pas si le titre aura une réelle importance, **utilisez des classes (comme toujours) et la balise <p>**
  - Vous changerez la balise le cas échéant

Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/HTML/Element/Heading\\_Elements](https://developer.mozilla.org/fr/docs/Web/HTML/Element/Heading_Elements)



**Questions ?**