



# La Plateforme

## / CSS - Les bonnes pratiques

## Sommaire



- Les unités relatives (Fluides)
- Les sélecteurs (class, id, naturel...) :
  - Quand utiliser l'un plutôt que l'autre ?
- Comment bien nommer ses classes

## Que mettre comme unités ? :

**Les unités fluides à disposition pour un site plus responsive sont :**

- **Em** : elle est proportionnelle à la taille de la police de l'élément parent ou du document. Par défaut, 1 em = 16 px si aucune taille de police n'est définie.
- **Rem** : l'unité rem fait toujours référence à la taille de la police de l'élément racine. En d'autres termes, elle dépend du font-size défini par défaut.
- **Les unités vh et vw** sont similaires, à la seule différence qu'elles dépendent respectivement de la hauteur et de la largeur de la fenêtre de navigation.
  - La hauteur du viewport (vh)
  - La largeur du viewport (vw)
  - Le viewport minimum (vmin)
  - Le viewport maximum (vmax)
- **Le pourcentage (%)** fait partie des unités relatives de façon générale puisqu'il s'adapte en fonction de son élément parent.
- La propriété **clamp(taille min, taille idéale, taille maximale)** : Nouveau en CSS3 ! l'élément ou le texte s'adaptera automatiquement à la taille !

**[Exemples à télécharger ICI](#)**

## Rappel des sélecteurs principaux :

- “**nom de l'élément**” pour sélectionner un élément

```
p {  
    ...  
}
```

- “.” Pour sélectionner une “classe”

```
.nom_class {  
    ...  
}
```

- “#” Pour sélectionner une “ID”

```
#nom_id {  
    ...  
}
```

```
.classe1 .classe2{  
    ...  
}
```

```
div > p {  
    ...  
}
```

```
input[type="text"] {  
    ...  
}
```

```
p.nom_class {  
    ...  
}
```

## Rappel des sélecteurs spéciaux :

Toute la liste de sélecteurs  
avec leur fonctionnement :

[CSS Selectors Reference](#)

## Rappel des sélecteurs : les pseudo-éléments

```
tr:nth-child(2) {  
    ...  
}
```

Sélectionne **le deuxième élément “tr”** rencontré dans la page

```
tr:nth-child(2n) {  
    ...  
}
```

Sélectionne **les “tr” paires** rencontrés dans la page

```
.ma-classe:nth-child(3){  
    ... }
```

Sélectionne **seulement le troisième élément qui possède la classe “ma-classe”** rencontré dans la page

```
li:first-child{  
    ... }
```

Sélectionne **le premier élément** d’une série d’éléments identiques et successifs

## Il existe d'autres pseudo-éléments

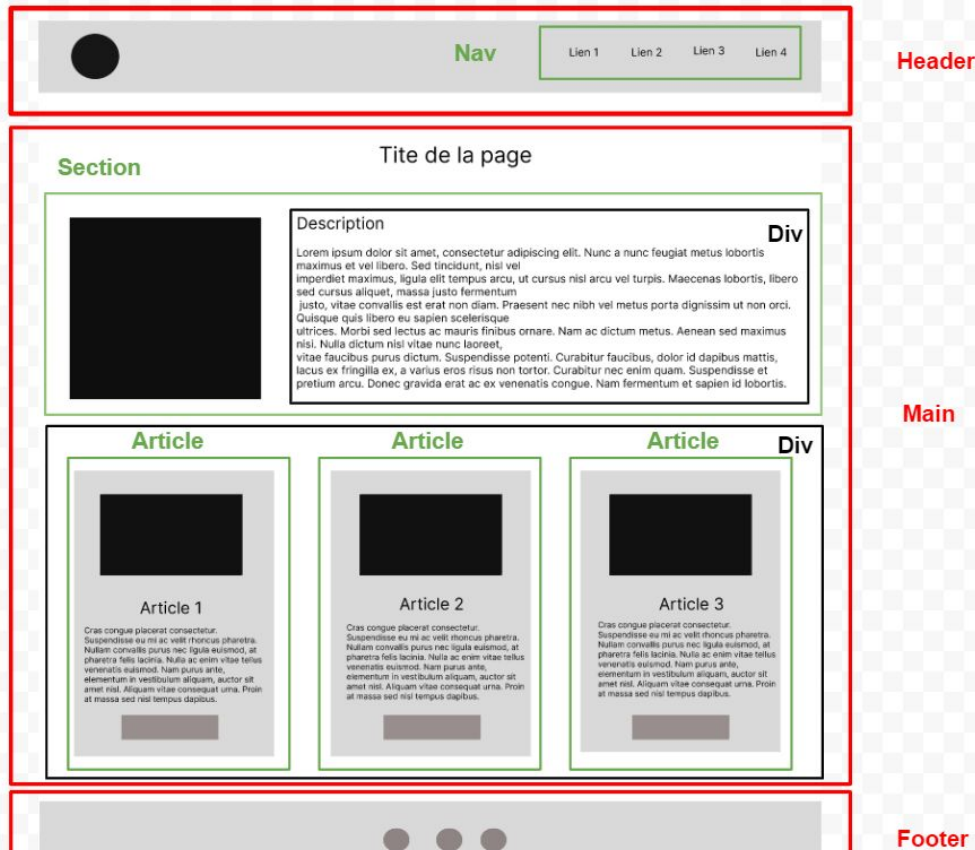
- ::after (:after)
- ::backdrop Expérimental
- ::before (:before)
- ::cue (:cue)
- ::first-letter (:first-letter)
- ::first-line (:first-line)
- ::grammar-error Expérimental
- ::marker Expérimental
- ::part Expérimental
- ::placeholder Expérimental
- ::selection
- ::slotted()
- ::spelling-error Expérimental

**[Pseudo-éléments - CSS : Feuilles de style en cascade | MDN](#)**

## Les sélecteurs en CSS :

## J'utilise lesquels et quand ?

**Exemples en LIVE  
sur la structure ci-contre :**





## Comment bien nommer ses classes ?

Il existe plusieurs “normes de nommage” officielles pour les classes et autres (BEM, OOCSS ...), mais la manière simple la plus utilisée et qui surtout différenciera vos noms de classes css des autres noms utilisés dans le dev est le **“KEBAB CASE”** :

mots en minuscule séparés par un tiret du 6 “-”.

Exemple : .container-articles , .bt-menu, .white-card, .contour-images, .centrer-vh ...

Le plus important est de :

- **Choisir une norme et s’y tenir.**
- **Être DRY (éviter les doublons de propriétés, réduire la taille du css ..)**

<https://www.alsacreation.com/article/lire/1641-Bonnes-pratiques-en-CSS-BEM-et-OOCSS.html>

# Questions ?