

# Introduction à CSS

Anna

## Qu'est-ce que le CSS ?

- CSS = Cascading Style Sheets
  - Langage de style qui décrit la présentation d'un document HTML
  - Permet de séparer le **contenu** (HTML) de la mise en **forme** (CSS)
- 

## Pourquoi utiliser CSS ?

- Changer les couleurs, polices, tailles
  - Créer des mises en page responsives
  - Améliorer l'accessibilité et l'esthétique
  - Réutiliser des styles à travers plusieurs pages
- 

## Syntaxe de base

```
sélecteur {  
  propriété: valeur;  
}
```

Exemple:

```
p {  
  color: blue;  
  font-size: 16px;  
}
```

---

## Les Selecteurs

- Par **balise**: `p { ... }`  
Le CSS affectera toutes les balises paragraphe `<p>`
  - Par **classe** : `.ma-classe { ... }` Peut s'appliquer à toutes les balises, flexible et réutilisable
  - Par **id** : `#mon-id { ... }` En soit la même chose que classe, mais pour cibler des éléments unique (souvent pour du JS ou aux ancras. Rarement pour le style)
  - **Combinés** : `div p { ... }`  
Affectera tous les `<p>` dans une `<div>`
- 

## Exemple de Selecteurs

### HTML

HTML :

```
<p>Par défaut</p>  
<p class="special">Avec une classe</p>  
<p id="unique">Avec un id</p>  
<div>  
  <p>Paragraphe dans un div</p>  
</div>
```

## CSS

CSS:

```
p {  
  color: blue;  
}  
  
.special {  
  font-weight: bold;  
  color: red;  
}  
  
#unique {  
  text-decoration: underline;  
}  
  
div p {  
  background-color: lightyellow;  
}
```

---

## Les propriétés principales

- **Couleurs:** color, background-color
- **Texte:** font-family, font-size, font-weight
- **Boîte (Box Model):** margin, padding, border
- **Disposition:** display, flex, grid, position

---

## Le modèle de boîte (Box Model)

1. content
2. padding
3. border
4. margin

## Bonnes pratiques

- Toujours séparer HTML et CSS (fichier `.css`)
  - Utiliser des classes plutôt que des ID's pour le style
  - Privilégier un style cohérent et réutilisable
  - Penser au **responsive design**
  - Si votre CSS n'apparaît pas en ligne: vider le cache ;)
- 

## Petites astuces et mauvaises pratiques

### Astuces utiles

- **Préfixes navigateurs** (encore visibles dans du vieux code) :

```
.box {  
  -webkit-border-radius: 10px; /* Chrome, Safari */  
  -moz-border-radius: 10px;    /* Firefox */  
  border-radius: 10px;         /* Standard */  
}
```

- Pour être sûre que les propriétés CSS soient supportées par le navigateurs (Chrome, Firefox, Safari, Edge, etc.) qui peut différer selon l'utilisateur
- Tu peux vérifier si tes propriétés CSS sont supportées en amont:
  - Spécifications W3C / WHATWG: Documentation officielle des standards du web, source la plus fiable, mais indigeste.
  - [Can I Use](#)
  - [MDN Web Docs](#)

### Mauvaises pratiques (mais sympa quand même)

- Abuser de `!important`:

```
p {  
  color: red !important; /* Override si un autre p color existe ailleurs*/  
}
```

- À réserver uniquement pour corriger un bug ponctuel.

---

## Les extensions de Bootstrap CSS

### Qu'est-ce que Bootstrap ?

- **Bootstrap** est un **framework CSS et JS** créé par Twitter en 2011.
- Il sert à **accélérer le développement web responsive**.
- Tu utilises des **classes toutes prêtes** au lieu d'écrire ton CSS à la main.

---

### Framework principal

- **Bootstrap**  
→ Framework CSS/JS d'origine (v5.3+).

---

### Extensions populaires

- **Bootstrap Social** → boutons stylés pour Facebook, Twitter, GitHub, etc.
- **Bootstrap Icons** → pack officiel d'icônes.
- **Bootstrap Select** → améliore les `<select>` en style Bootstrap.
- **Bootstrap DatePicker / Timepicker** → champs calendrier/heure stylés.
- **Bootstrap Table** → tableaux avec pagination, tri, etc.
- **Bootstrap Tags Input** → champs de saisie avec tags.

## Thèmes et distributions

- [Bootswatch](#) → collection de thèmes gratuits.
  - **MDBootstrap** (**Material Design for Bootstrap**) → design basé sur Google Material.
  - **Flatly**, **Cosmo**, **Lumen**, **Darkly**... (via Bootswatch).
- 

## Intégrations front-end

- **React-Bootstrap** → composants React basés sur Bootstrap.
  - **NG-Bootstrap** → intégration Angular.
  - **BootstrapVue** → intégration Vue.js.
- 

## En résumé Pour Bootstrap

- **Bootstrap** → la base.
  - **Plugins** → Social, Select, Datepicker, etc.
  - **Intégrations** → React, Angular, Vue.
  - **MDBootstrap** → un dérivé très populaire avec Material Design.
- 

## Ressources utiles CSS

- [MDN Web Docs – CSS](#)  
La référence officielle pour comprendre chaque propriété.
- [CSS Tricks](#)  
Articles et astuces pour progresser en CSS.

- [Flexbox Froggy](#)  
Jeu interactif pour apprendre **Flexbox**.
  - [Grid Garden](#)  
Jeu interactif pour apprendre **CSS Grid**.
  - ChatGPT
- 

**That's all Folks!**

by Anna.