



CSS

Introduction

Objectifs Pédagogiques

À l'issue de cette journée, vous serez en mesure de :

- ✓ Comprendre les fondamentaux du langage CSS
- ✓ Comprendre la notion de sélecteur
- ✓ Connaître les propriétés de base

Langage CSS

CSS = Cascading Style Sheets (feuilles de style en cascade)

Une feuille de style vient en complément d'une page HTML.

L'idée est de séparer les responsabilités :

- **HTML** => sémantique (**fond**).
- **CSS** => mise en **forme**.

Utilisation CSS

Des règles CSS peuvent s'appliquer :

- à un élément via l'attribut **style**.

```
<h2 style="color: red;">Titre</h2>
```

- dans une page HTML via l'élément **<style>**

```
<style>
  h2 {
    color: red;
  }
</style>
```

- via un fichier externe (.css) référence dans une page HTML via la balise **<link>**

```
<link rel="stylesheet" href="css/style.css">
```

Syntaxe générale

Une feuille de style contient des règles de la forme :

```
selecteur(s) {  
    propriété: valeur;  
}
```

Les commentaires

```
/*  
Ceci est un commentaire  
*/
```

Exemple :

```
/*  
Tous les titres de niveau 2 auront  
la couleur rouge.  
*/  
h2 {  
    color: red;  
}
```

Sélecteurs multiples

Les sélecteurs sont séparés par des virgules.

```
selecteur1, selecteur2, selecteurN {  
  propriété: valeur;  
}
```

Exemple :

```
div, p, footer {  
  color: red;  
}
```

Propriétés combinées

Les sélecteurs sont séparés par des virgules.

```
selecteur {  
  propriété: valeur1 valeur2 valeur3;  
}
```

Exemple :

```
td {  
  border: 1px solid black;  
}
```

Sélecteurs

Sélecteurs simples

Sélecteur	Description
*	Tous les éléments.
élément	Tous les éléments « élément »
#identifiant	L'élément avec id ="identifiant"
élément.uneClasse	Le ou les éléments « élément » avec class ="uneClasse"
.uneClasse	Tous les éléments avec class ="uneClasse"
élément1 élément2	Tous les « élément2 » contenus dans l'élément « élément1 »
élément1 > élément2	Tous les « élément2 » ayant pour parent l'élément « élément1 »
élément1 + élément2	Tous les « élément2 placés juste après l'élément « élément1 »
élément1 ~ élément2	Tous les « élément2 » précédés par l'élément « élément1 »

Sélecteurs d'attributs

Sélecteur	Description
[attribut]	Tous les éléments avec l'attribut « attribut ».
[attribut=valeur]	Tous les éléments dont la valeur de « attribut » vaut « valeur ».
[attribut*=valeur]	Tous les éléments dont la valeur de « attribut » contient « valeur ».
[attribut^=valeur]	Tous les éléments dont la valeur de « attribut » commence par « valeur ».
[attribut\$=valeur]	Tous les éléments dont la valeur de « attribut » finit par « valeur ».

Pseudo-classes structurales

Sélecteur	Description
élément:first-child	Tous les « élément » premier enfant de leur parent.
élément:last-child	Tous les « élément » dernier enfant de leur parent.
élément:nth-child(n)	Tous les « élément » qui sont n-ième enfant de leur parent.
élément:nth-last-child(n)	Tous les « élément » qui sont n-ième enfant de leur parent à partir de la fin .
élément:first-of-type	Tous les « élément » premier enfant de type « élément » de leur parent.
élément:last-of-type	Tous les « élément » dernier enfant de type « élément » de leur parent.
élément:only-child	Tous les « élément » seul enfant de leur parent.
élément:only-of-type	Tous les « élément » seul enfant de type « élément » de leur parent.
élément:not(sélecteur)	Tous les « élément » ne correspondent pas au sélecteur.

Autres Pseudo-classes

Sélecteur	Description
élément:active, élément:focus, élément:hover	Tous les « élément » durant les actions utilisateurs.
élément:link	Tous les « élément » cible d'un hyperlien pas encore visité.
élément:visited	Tous les « élément » cible d'un hyperlien déjà visité.
élément:checked	Tous les « élément » cochés (actions de l'utilisateur).
élément:before	Ajoute du contenu avant « élément ».
élément:after	Ajoute du contenu après « élément ».
élément:first-line	Première ligne de chaque élément.
élément:first-letter	Première lettre formatée de chaque « élément ».

Propriétés

Police

Propriétés de police

font-family: liste_de_noms

font-style: normal | italic | oblique

font-variant: normal | small-caps

font-weight: normal | bold | bolder | lighter

font-size: taille

La famille de police sera choisie

En fonction des polices disponibles

Dans l'ordre d'apparition dans la liste

@font-face définit une police personnalisée

```
@font-face {  
  font-family: MaPolicePerso;  
  src: url(MaPolicePerso.woff)  
}
```

```
p {  
  font-family: Helvetica;  
  font-size: 24px;  
  font-weight: bold;  
  font-style: italic;  
}
```

Texte

Quelques propriétés du texte

text-decoration: underline | overline | line-through | blink

text-transform: capitalize | uppercase | lowercase | none

text-align: left | right | center | justify

Quelques propriétés de positionnement du texte

vertical-align: sub | super | baseline | middle | text-top | text-bottom

line-height: hauteur_boite_ligne

```
p {  
    text-align: right;  
    text-decoration: underline;  
    text-transform: capitalize;  
    line-height: 3;  
}
```

Modèle de boîte

Chaque élément du document à présenter est modélisé par une boîte rectangulaire agencée selon des propriétés de formatage?

Une boîte possède

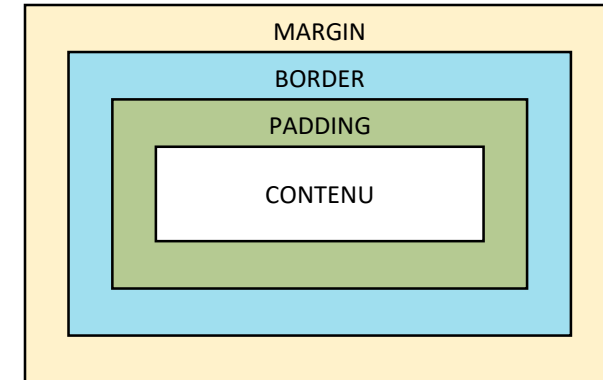
- une zone de contenu
- des zones de remplissage (propriété **padding**)
- des zones de bordure (propriété **border**)
- des zones de marge (propriété **margin**)

Les propriétés peuvent être suffixées par **-top**, **-left**, **-bottom**, **-right** pour préciser la zone concernée.

Par exemple : **padding-top**

Exemple

```
p {  
  padding-top: 10px;  
  border: solid 1px;  
  margin: 10px;  
}
```



Fond

Propriété	Description
background-color	Couleur de fond d'un bloc.
background-image	Lien d'une image à mettre en fond.
background-repeat	Répétition du fond
background-size	Taille de l'image de fond.
background-position	Position d'origine du fond.

```
body {  
    background-image : url(https://upload.wikimedia.org/wikipedia/commons/thumb/6/61/HTML5_logo_and_wordmark.svg/240px-HTML5_logo_and_wordmark.svg.png);  
    background-repeat: no-repeat;  
    background-position: right top;  
}
```

Couleurs

Propriété	Description
color	Couleur de texte
border-color	Couleur de la bordure d'un bloc
background-color	Couleur de fond d'un bloc

display

La propriété **display** définit le type d'affichage utilisée pour le rendu d'un élément (de bloc ou en ligne) et la disposition utilisée pour ses éléments fils : grille ou boîtes flexibles.

Exemple de valeurs :

display: none => l'élément n'est pas affiché.

display: block => l'élément se positionne en bloc. L'élément suivant passe à la ligne.

display: inline => l'élément se positionne en ligne. L'élément suivant peut être sur la même ligne.

...

Bordures

La propriété CSS **border** est une propriété raccourcie qui permet de définir les propriétés liées à la bordure. **border** peut être utilisée pour définir les valeurs de **border-width**, **border-style** et **border-color**.

Syntaxe :

```
/* style */
```

```
border: solid;
```

```
/* largeur | style */
```

```
border: 2px dotted;
```

```
/* style | couleur */
```

```
border: solid #f33;
```

```
/* largeur | style | couleur */
```

```
border: 2px dashed green;
```

border-radius

La propriété **border-radius** permet de définir des coins arrondis pour la bordure d'un élément

Syntaxe :

```
/* Quand on utilise un seul rayon, on peut avoir jusqu'à quatre valeurs */  
/* On utilise le même rayon pour les quatre angles */  
border-radius: 10px;
```

```
/* 1. coin en haut à gauche */  
/* 2. coin en haut à droite */  
/* 3. coin en bas à droite */  
/* 4. coin en bas à gauche */  
border-radius: 1px 0 3px 4px;
```

Pour aller plus loin : <https://developer.mozilla.org/fr/docs/Web/CSS/border-radius>

list-style-type

La propriété **list-style-type** permet de définir l'apparence de la puce d'un élément de liste (un cercle, un caractère, un compteur spécifique, etc.).

Exemple :

```
list-style-type: disc;  
list-style-type: circle;  
list-style-type: square;  
list-style-type: none;
```

Pour aller plus loin : <https://developer.mozilla.org/fr/docs/Web/CSS/list-style-type>

Travaux Pratiques

Créer un dépôt workshop-css.

Réaliser les TP01 et TP02.

Créer une page index.html qui contient les liens vers les pages réalisées.

Publier vos modifications dans GitHub Pages.