

## Liste des propriétés CSS (les plus utilisées)

La liste complète des propriétés CSS (avec démos interactives) est disponible ici :  
<https://developer.mozilla.org/fr/docs/Web/CSS/Reference>

Effet	Propriété	Valeur
<b>Arrière-plan</b>		
Définir la couleur de fond	background-color	<couleur>
		transparent
Définir l'image de fond	background-image	<lien>
Définir la taille de l'image de fond	background-size	<taille> cover contain auto
<b>Bordure</b>		
Définir la couleur de la bordure inférieure/supérieure gauche/droite	border-bottom-color border-top-color border-left-color border-right-color	<couleur>
Définir le style de la bordure inférieure/supérieure gauche/droite	border-bottom-style border-top-style border-left-style border-right-style	dotted dashed solid ...
Définir l'épaisseur de la bordure inférieure/supérieure gauche/droite	border-bottom-width border-top-width border-left-width border-right-width	<taille> thin medium thick
Définir complètement la bordure	<b>border</b>	<border-width> <border-style> <border-color>
Définir l'épaisseur de la bordure (4 cotés)	border-width	<taille> thin medium thick
Définir la couleur de la bordure (4 cotés)	border-color	<couleur>
Définir les coins arrondis de la bordure (4 cotés)	<b>border-radius</b>	<taille> <taille> <taille> <taille>
Définir le style de la bordure (4 cotés)	border-style	dotted dashed solid

		...
Ajouter des ombres	box-shadow	<taille> <taille> <taille> <taille> <couleur>
Détermine si les bordures du tableau sont fusionnées ou séparées	border-collapse	collapse separate
<b>Police</b>		
Définir la famille de police	font-family	<police>
Définir la taille de la police	font-size	<taille>
Sélectionner une fonte italique ou oblique	font-style	italic oblique ...
Définir des paramètres typographiques	font-variant	normal small-caps
Définir la graisse	font-weight	bold bolder lighter ...
<b>Texte</b>		
Définir l'espace entre les lettres	letter-spacing	<taille>
Définir l'alignement	text-align	start end left right center justify
Définir des décorations	text-decoration	none underline overline line-through blink
Appliquer des transformations	text-transform	none capitalize uppercase lowercase
Définir l'espace entre les mots	word-spacing	<taille>
<b>Couleur</b>		



Indique si un élément peut être situé à côté d'éléments flottants qui le précèdent ou s'il doit être déplacé vers le bas pour être en dessous de ces éléments	clear	left right both none
Définir la hauteur du conteneur	height	auto <taille>
Définir la largeur du conteneur	width	auto <taille>
Définir la hauteur maximale/minimale d'un conteneur	max-height min-height	none <taille>
Définir la largeur maximale/minimale d'un conteneur	max-width min-width	none <taille>
Définir la taille des marges sur les 4 côtés du conteneur	margin	<margin-top> <margin-right> <margin-bottom> <margin-left>
Définir la taille des marges sur le côté inférieur/gauche/droit/supérieur	margin-bottom margin-left margin-right margin-top	auto length
Définir les écarts de remplissage sur les 4 côtés du conteneur	padding	<padding-top> <padding-right> <padding-bottom> <padding-left>
Définir l'écart des remplissages sur le côté inférieur/gauche/droit/supérieur	padding-bottom padding-left padding-right padding-top	length
Définir comment gérer le dépassement du contenu d'un élément dans son conteneur	overflow-x overflow-y	visible hidden scroll auto
Cacher un élément tout en conservant l'espace occupé dans lequel il aurait été visible	visibility	visible hidden collapse

# Liste des sélecteurs CSS

La liste complète des sélecteurs CSS est disponible ici :

[https://developer.mozilla.org/fr/docs/Learn/CSS/Building\\_blocks/Selectors](https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Selectors)

Nom du sélecteur	Informations	Exemple(s)
Universel	Tous les éléments	<code>* { ... }</code>
Type	Tous les éléments de ce type	<code>h1 { ... }</code>
Groupe	Tous les éléments parmi ces balises	<code>h1, h2, h3 { ... }</code>
Classe	Plusieurs éléments de balises différentes sans forcément tous les sélectionner	<code>.maClasse { ... }</code> <code>p.maClasse { ... }</code>
Identifiant	Un seul élément en particulier	<code>#monId { ... }</code> <code>div#monId { ... }</code>
Descendant	Un élément qui se trouve dans un autre élément (peu importe le niveau)	<code>.maClasse h1 { ... }</code>
Enfant	Un élément qui se trouve dans un autre élément et qui est son enfant direct	<code>.maClasse &gt; h1 { ... }</code>
Frère adjacent	Tous les éléments qui possèdent le même parent et qui sont adjacents	<code>h1 + p { ... }</code>
Frère général	Tous les éléments qui possèdent le même parent	<code>h1 ~ p { ... }</code>
Pseudo-classe <sup>1</sup>	Tous les éléments qui possèdent cette pseudo-classe	<code>p:first-child { ... }</code> <code>p:last-child { ... }</code> <code>p:nth-child(3) { ... }</code> <code>p:hover { ... }</code>
Attribut	Tous les éléments qui possèdent cet attribut	<code>img[height="100"] { ... }</code> <code>img[src*=".png"] { ... }</code> <code>input[type="text"] { ... }</code>

---

<sup>1</sup> La liste des pseudo-classes est disponible à la page suivante

Quand un groupe de déclarations s'applique à plusieurs éléments distincts, on peut combiner les sélecteurs en une liste. Par exemple, si l'on souhaite cibler les paragraphes et les titre 1 pour qu'ils s'affichent en rouge, on pourrait écrire :

```
p {  
  color: red  
}  
  
h1 {  
  color: red  
}
```

Mais dans ce cas, il est préférable de combiner les sélecteurs en une seule règle, en les séparant par une virgule :

```
p, h1 {  
  color: red  
}
```

## Liste des pseudo-classes CSS

(les plus utilisées)

La liste complète des pseudo-classes est disponible ici : <https://developer.mozilla.org/fr/docs/Web/CSS/Pseudo-classes>

Pseudo-classe	Information	Exemple
:active	Cible un élément lorsque celui-ci est activé par l'utilisateur	<pre>/* ne cible &lt;a&gt; que lorsqu'il est activé */ /* par exemple quand on clique dessus */ a:active {   color: red; }</pre>
:checked	Cible un bouton radio, case à cocher ou option qui est coché ou activé par l'utilisateur	<pre>/* cible n'importe quel bouton radio sélectionné, case à cocher cochée ou option sélectionnée */ input:checked {   margin-left: 25px; }</pre>
:first-child	Cible un élément qui est le premier enfant de son parent	<pre>/* Cible n'importe quel élément &lt;p&gt; qui est */ /* le premier fils de son élément parent */ p:first-child {   color: lime; }</pre>
:first-of-type	Cible le premier élément d'un type donné parmi ceux d'un même élément parent	<pre>/* Cible le premier élément &lt;p&gt; d'un type donné */ /* parmi ses éléments voisins */ p:first-of-type {   color: red; }</pre>
:hover	Cible un élément qui est survolé par le pointeur de la souris	<pre>/* Cible n'importe quel élément &lt;a&gt; lorsque */ /* celui-ci est survolé */ a:hover {   background-color: gold; }</pre>

		<pre> }</pre>
:last-child	Cible un élément qui est le dernier enfant de son parent	<pre> /* Cible n'importe quel élément &lt;li&gt; tant que */ /* celui-ci est le dernier enfant de son élément */ /* parent */ li:last-child {     background-color: lime; }</pre>
:last-of-type	Cible un élément qui est le dernier enfant d'un type donné dans la liste des enfants de l'élément parent	<pre> /* Cible n'importe quel paragraphe qui est */ /* le dernier paragraphe de son élément parent */ p:last-of-type {     color: lime; }</pre>
:nth-child	Cible un élément qui est le nième enfant de son parent	<pre> /* Cible n'importe quel paragraphe qui est */ /* le 2ième paragraphe de son élément parent */ p:nth-child(2) {     background-color: lightblue; }</pre>
:link	Cible les liens qui n'ont pas encore été visités	<pre> /* Cible les liens qui n'ont pas encore */ /* été visités */ a:link {     color: red; }</pre>
:visited	Cible les liens qui ont été visités	<pre> /* Cible les liens qui ont été visités */ a:visited {     color: red; }</pre>



