

Largeur et centrage

Un texte qui court sur toute la largeur d'un grand écran est illisible. Voici comment limiter une boîte, la centrer, et lui donner une allure soignée.

4TTR

 Découverte

Ouvre n'importe quel site d'information sur un grand écran. Le texte n'occupe jamais toute la largeur : il reste dans une colonne centrale, avec du vide de chaque côté.

Ce n'est pas de la coquetterie. Au-delà d'une certaine longueur de ligne, l'œil ne retrouve plus le début de la ligne suivante et la lecture devient fatigante.



Objectifs

À la fin de ce cours, tu seras capable de :

1. Choisir entre `width` et `max-width`.
2. Centrer un bloc dans la page.
3. Distinguer « centrer la boîte » de « centrer le texte ».
4. Arrondir les angles et poser une ombre.

Limiter la largeur

Deux propriétés se ressemblent, et une seule est presque toujours le bon choix.

Propriété	Effet
<code>width: 400px</code>	Largeur imposée : 400 px, même dans une fenêtre de 320 px
<code>max-width: 400px</code>	Largeur maximale : 400 px, ou moins si la fenêtre est plus étroite

```
1 | .intro {  
2 |     max-width: 400px;  
3 | }
```

Avec `width`, un téléphone doit faire défiler la page horizontalement pour lire ton texte. Avec `max-width`, la boîte se réduit d'elle-même. C'est la première brique de ce qu'on appelle le **GB responsive design** — une page qui

s'adapte à la taille de l'écran.

💡 **ASTUCE** : pour du texte courant, une largeur maximale entre 600 et 800 pixels donne des lignes confortables. Au-delà, la lecture se dégrade.

Centrer un bloc

Voici l'astuce la plus utile du CSS, et la moins devinable :

```
1 | .intro {  
2 |     max-width: 400px;  
3 |     margin-left: auto;  
4 |     margin-right: auto;  
5 | }
```



`auto` veut dire « le navigateur calcule ». En lui demandant la même chose à gauche et à droite, il ne peut faire qu'une seule chose : partager l'espace restant en deux parts égales. La boîte se retrouve au milieu.

Deux conditions, sans lesquelles ça ne marche pas :

- la boîte doit être un élément **bloc** (un `div`, un `main`, un `p` — pas un `span`) ;
- elle doit avoir une **largeur limitée**. Une boîte qui prend déjà toute la largeur n'a aucun espace à partager.

Le raccourci

En pratique, on écrit rarement les deux lignes. Le raccourci `margin` fait le travail :

```
1 | .intro {  
2 |     max-width: 400px;  
3 |     margin: 30px auto;  
4 | }
```

Première valeur pour le haut et le bas, seconde pour la gauche et la droite. Donc : 30 px au-dessus et en dessous, et `auto` sur les côtés — c'est-à-dire centré.

Centrer la boîte, ou centrer le texte ?

Deux choses très différentes portent le même nom en français, et c'est une source d'erreurs permanente.

Écriture	Ce qui bouge
<code>margin: auto</code>	La boîte se place au milieu de la page
<code>text-align: center</code>	Le contenu se place au milieu de la boîte

```
1 | header {  
2 |   text-align: center;  
3 | }
```

Ici, l'en-tête garde toute sa largeur — c'est son titre qui vient au centre.

⚠ **ATTENTION** : `text-align: center` s'écrit sur la boîte **parente**, pas sur le texte. Tu centres le contenu d'une boîte, jamais un élément lui-même.

Enlever la marge du body

Le navigateur applique par défaut une marge d'environ 8 pixels autour du `<body>`. C'est ce liseré blanc que tu vois tout autour d'une page, y compris derrière un en-tête coloré.

```
1 | body {  
2 |   margin: 0;  
3 | }
```

Sur un vrai site, on l'enlève presque toujours : un bandeau d'en-tête doit toucher les bords de la fenêtre, sinon il flotte bizarrement.

Les finitions

Deux propriétés qui ne servent à rien de fonctionnel, et qui changent pourtant tout à l'allure d'une page.

```
1 | .carte {  
2 |   border-radius: 10px;  
3 |   box-shadow: 0 2px 8px rgba(0, 0, 0, 0.15);  
4 | }
```

`border-radius` arrondit les coins. Plus la valeur est grande, plus c'est rond — jusqu'à `50%`, qui transforme un carré en cercle. C'est comme ça qu'on fait les photos de profil rondes.

`box-shadow` pose une ombre portée. Ses quatre valeurs sont, dans l'ordre : le décalage horizontal, le décalage vertical, le flou, et la couleur.

💡 **ASTUCE** : pour une ombre crédible, garde le décalage horizontal à `0` et un flou généreux. Une ombre nette et décalée en diagonale fait tout de suite « années 2000 ».



Vérifications

1. Quelle propriété choisis-tu pour un bloc de texte, et pourquoi ?
2. Pourquoi `margin: auto` ne centre-t-il rien sur une boîte sans `max-width` ?
3. Que fait `margin: 20px auto;` exactement ?
4. Tu veux centrer le titre à l'intérieur d'un en-tête qui occupe toute la largeur. Quelle propriété ?



À retenir

- `max-width` plutôt que `width` : la boîte se réduit sur les petits écrans.
- La recette du centrage : une **largeur limitée** + `margin-left: auto` et `margin-right: auto`.
- Le raccourci `margin: 30px auto;` fait les deux d'un coup.
- `margin: auto` centre **la boîte** ; `text-align: center` centre **son contenu**.
- `body { margin: 0; }` supprime le liseré blanc par défaut.

Suite

Tes boîtes sont bien placées, mais ce sont toutes des `<div>` anonymes. HTML propose des conteneurs qui, eux, portent un nom.