

Les images en HTML (tutoriel)

Remarque. Ce tutoriel est extrait de la section de MDN intitulée : [Les images en HTML](#)

Au début, le Web n'était que du texte, ce qui était un peu ennuyeux. Heureusement, il n'a pas fallu longtemps pour que la possibilité d'intégrer des images (et d'autres types de contenu intéressants) dans une page web soit ajoutée. Bien qu'il y ait plusieurs types de contenu multimedia, il est logique de commencer avec l'humble élément ``, utilisé pour intégrer une image dans une page web. Dans cet article, nous approfondirons son utilisation en abordant les principes fondamentaux, l'annotation par légendes utilisant `<figure>`, et en analysant sa relation avec les images d'arrière-plan du CSS.

Comment intégrer une image à une page web ?

Pour mettre une image simple sur une page web, nous utiliserons l'élément ``. C'est un **élément vide** (ce qui signifie qu'il ne contient ni texte ni balise de fermeture) qui demande au moins un attribut pour fonctionner — `src` (souvent appelé par son nom entier: *source*). L'attribut `src` contient un chemin pointant vers l'image que vous voulez intégrer, qui peut être une URL absolue ou relative, de la même manière que l'élément `<a href=` attribue des valeurs.

Donc, par exemple, si votre image s'appelle `dinosaur.jpg`, et qu'elle est située dans le même répertoire que votre page HTML, vous pouvez intégrer cette image comme ceci (URL relative) :

```

```

Et si cette image se trouve dans un sous-répertoire `images` situé dans le même dossier que la page HTML (ce que Google recommande dans un but d'indexation et d'optimisation de la recherche), alors vous l'intégrerez comme ceci :

```

```

Vous pouvez intégrer l'image en utilisant son URL absolue, par exemple :

```

```

Le code au-dessus vous donnera, à peu près, le résultat suivant :



Note : Vous trouverez les exemples finis de cette section sur [Github](#) (regardez aussi le [code source](#).)

Texte alternatif

Le prochain attribut que nous allons étudier est `alt`. Sa valeur est supposée être un descriptif sous forme de texte de l'image, à utiliser dans les cas où l'image ne peut être affichée. Exemple : le code au-dessus pourrait être modifié de cette manière :

```

```

La manière la plus simple de tester votre texte `alt` est de mal épeler votre nom de fichier intentionnellement. Si dans l'exemple, la photo était épelée `dinosooooor.jpg`, le navigateur ne l'afficherait pas mais afficherait le texte `alt` à la place.

Pourquoi vous verrez partout du texte `alt` ? Vous en aurez besoin car c'est très pratique en maintes occasions :

- L'utilisateur est un déficient visuel qui utilise un [lecteur d'écran](#) qui s'en sert pour "lire" le web. En fait, avoir du texte `alt` disponible pour décrire les images est très utile à beaucoup d'utilisateurs.
- Comme nous l'avons vu au-dessus, vous pourriez avoir mal épelé le nom ou le chemin du fichier.

- Le navigateur ne gère pas ce type d'image. Certains utilisent encore des navigateurs en terminal, affichant seulement du texte (comme [Lynx](#)), qui affichent le texte alt des images.
- Vous pouvez avoir envie de fournir du texte que pourraient utiliser les moteurs de recherche. Par exemple, ils mettront en relation le texte alt avec des requêtes de recherche.
- L'utilisateur a supprimé les images pour économiser le volume du transfert de données ou pour ne pas être distrait. C'est très commun sur les téléphones mobiles et dans les pays où la bande passante est limitée ou coûte cher.

Que devriez-vous noter dans vos attributs `alt` ? En premier lieu, cela dépend de la raison pour laquelle cette image se trouve là. En d'autres mots, ce que vous perdriez si cette image ne s'affichait pas :

- **Decoration.** Vous devriez utiliser [des images d'arrière-plan CSS](#) pour les images décoratives mais si vous devez utiliser du HTML, ajoutez un `alt=""` vide. Si l'image ne fait pas vraiment partie du contenu, un lecteur d'écran ne perdra pas de temps à la lire.
- **Contenu.** Si votre image fournit une ou plusieurs informations supplémentaires significatives, inscrivez ces mêmes informations dans un *brief alt text* – ou mieux, dans le texte principal, que tout le monde puisse les voir. N'écrivez pas de `alt text` redondants. Imaginez combien ce serait ennuyeux pour un lecteur si tous les paragraphes étaient écrits en double... Si l'image est décrite de manière adéquate dans le corps de texte principal, vous pouvez utiliser simplement `alt=""`.
- **Lien.** Si vous mettez une image à l'intérieur d'une ancre `<a>` pour transformer une image en lien, vous devez quand même fournir un [Lien texte accessible](#). Dans de tels cas, vous pouvez, soit l'inclure dans le même élément `<a>`, soit dans l'attribut `alt` de l'image – utilisez ce qui marche le mieux dans votre cas.
- **Texte.** Vous ne devez pas mettre de texte dans les images. Si votre titre principal a besoin d'un peu d'ombrage par exemple, [utilisez CSS](#) pour ça, plutôt que de mettre du texte dans une image. De toutes manières, si vous ne pouvez pas éviter de faire ça, vous devez ajouter le texte dans l'attribut `alt` .

Le but est de livrer essentiellement une expérience de qualité, même quand les images ne peuvent être vues. Cela assure à tous les utilisateurs de ne rien manquer du contenu. Essayez de ne pas afficher les images dans votre navigateur et regardez ce qu'il se passe. Vous allez vite réaliser que le texte fourni à la place est réellement utile.

Largeur et hauteur (`width-height`)

Vous pouvez vous servir des attributs `width` et `height` pour spécifier la largeur et la hauteur de votre image. Vous pouvez trouver la largeur et la hauteur de différentes manières. Sur Mac, par exemple, vous pouvez utiliser `Cmd + I` pour afficher l'info relative au fichier image. Pour revenir à notre exemple, nous pourrions faire ceci :

```

```

C'est une bonne pratique, cela donne une page se chargeant plus rapidement et en douceur.

De toutes manières, vous ne devez pas altérer la taille de vos images avec les attributs HTML . Si vous réglez la taille de l'image trop grande, vous aurez un résultat avec beaucoup de "grain", flou ou trop petit et vous dépensez de la bande passante en téléchargeant une image qui ne convient pas aux besoins de l'utilisateur. Votre image peut aussi sortir distordue, si vous n'en maintenez pas le bon [Format d'image](#). Vous devriez utiliser un éditeur d'images pour la mettre à la bonne taille avant de la mettre dans votre page web.

Note : Si vous devez absolument modifier une taille d' image, vous devriez vous servir de [CSS](#) .

Titre d'images

Comme décrit dans le chapitre [Création d'hyperliens](#), vous pouvez aussi ajouter un attribut `title` aux images, pour fournir un supplément d'information si nécessaire. Dans notre exemple, nous pourrions faire ceci :

```

```

Cela donne une info-bulle avec le texte entré dans l'attribut `title` :



Il n'est pas essentiel d'inclure des informations dans les images. Il est souvent préférable d'écrire ces informations dans le texte principal plutôt qu'attaché à l'image. Ils peuvent être très utile dans d'autres circonstances, par exemple dans une galerie d'images où vous n'avez pas de place pour les légendes.

Légender des images avec `figure` et `figcaption`

En parlant de légendes, il y a de nombreuses manières d'en ajouter qui ira avec votre image. Par exemple, rien ne vous empêche de faire ceci :

```
<div class="figure">
  

  <p>A T-Rex on display in the Manchester University Museum.</p>
</div>
```

C'est bon. Ça contient ce que vous voulez et c'est aisément stylisable en CSS. Mais il y a un problème : il n'y a rien de sensé qui relie l'image à sa légende. Ce qui peut poser des problèmes à un lecteur d'écran. Par exemple, quand vous avez 50 images, quelle légende va avec quelle image ?

Une meilleure solution consiste en l'utilisation des éléments HTML5 `<figure>` et `<figcaption>`. Ils ont été conçus pour cela : fournir un conteneur sémantique aux objets et lier clairement cet objet à sa légende. Notre exemple précédent pourrait être réécrit comme ceci :

```
<figure>
  

  <figcaption>
    A T-Rex on display in the Manchester University Museum.
  </figcaption>
</figure>
```

L'élément `<figcaption>` dit au navigateur et aux technologies d'assistance que la légende décrit le contenu de l'autre élément `<figure>`.

Un objet `<figure>` n'est pas forcé de contenir une image. C'est une unité de contenu indépendante qui :

- exprime votre désir d'accessibilité et de compréhension facilitée.
- peut se placer en plusieurs endroits dans une page à flot linéaire.
- Fournit une information essentielle qui supporte le texte principal.

Cet objet peut être un ensemble d'images, des bribes de code, de l'audio, de la vidéo, des équations, un tableau ou bien d'autres choses.