

HTML sémantique

HTML sémantique

Le HTML sémantique introduit un sens au code que nous écrivons. Avant le HTML sémantique, les éléments n'avaient aucune signification quant à ce qu'il fait ou au contenu qu'il contient. Un élément tel que `<div>` a été utilisé comme élément à usage général pour créer des éléments allant des en-têtes aux pieds de page en passant par les articles. Avec Semantic HTML, nous avons découvert des éléments qui indiquent aux développeurs et aux navigateurs exactement ce qu'il fait et quel contenu doit y figurer.

```
<!--Non Semantic HTML-->
<div id="footer">
  <p>this is a footer</p>
</div>
```

```
<!--Semantic HTML-->
<footer>
  <p>this is a footer</p>
</footer>
```

Placement des éléments

Le HTML sémantique introduit des éléments qui peuvent dire aux développeurs exactement ce que fait l'élément ou où il est placé en fonction du nom de cet élément. Certains de ces éléments sont `<header>`, `<nav>`, `<main>` et `<footer>`. `<header>` décrit le contenu en haut de la page `<body>`. Il peut inclure un logo, des liens de navigation ou une barre de recherche. `<nav>` encapsule les liens de navigation de la page. Il est souvent placé à l'intérieur du `<header>` ou `<footer>`. `<main>` encapsule le contenu principal d'une page entre l'en-tête/la navigation et les zones de pied de page. `<footer>` inclut le contenu du pied de page de la page au bas du fichier `<body>`.

Intégration des médias

Le HTML sémantique nous présente `<video>`, `<audio>` et `<embed>`. `<video>` nous permet d'ajouter des vidéos à notre site Web. `<audio>` nous permet de mettre en œuvre l'audio dans notre site Web. `<embed>` peut être utilisé pour mettre en œuvre tout type de média. Ces éléments sont universels dans la mesure où ils utilisent tous l'attribut `src` pour lier la source du contenu. `<video>` et `<audio>` nécessitent une balise de fermeture alors qu'il `<embed>` s'agit d'une balise à fermeture automatique.

```
<!--Video Tag-->
<video src="4kvideo.mp4">video not
supported</video>
```

```
<!--Audio Tag-->
<audio src="koreanhiphop.mp3">
</audio>
```

```
<!--Embed tag-->
<embed src="babyyoda.gif"/>
```

<figure> et <figcaption>

L' `<figure>` élément est utilisé pour encapsuler des médias tels qu'une image, un diagramme. ou un extrait de code. L' `<figcaption>` élément est utilisé pour décrire le média encapsulé dans l' `<figure>` élément. Les développeurs utiliseront normalement `<figcaption>` dans l' `<figure>` élément pour regrouper le média et la description. De cette façon, si un développeur décide de changer la position du média, la description suivra.

<section> et <article>

`<section>` définit des éléments dans un document, tels que des chapitres, des titres ou toute autre zone du document ayant le même thème.

`<article>` contient du contenu qui a du sens en soi, comme des articles, des blogs et des commentaires.

Généralement, les développeurs utiliseront

`<section>` pour définir un thème pour la page Web et `<article>` pour écrire du contenu indépendant pour ce thème. Cela ne signifie pas qu'il `<article>` doit être utilisé avec `<section>` .

<aside> Élément de côté

L' `<aside>` élément est utilisé pour marquer des informations supplémentaires qui peuvent améliorer un autre élément mais ne sont pas nécessaires pour comprendre le contenu principal. Habituellement, ces informations se trouvent dans une barre latérale ou à un endroit où elles n'obstruent pas le contenu principal. Un exemple de ceci serait un article qui explique comment prendre soin d'un chien et à côté de l'article une annonce apparaîtrait faisant la publicité d'un produit de toilettage pour chiens.

```
<figure>
  
  <figcaption>The image shows the
  layout of a qwerty keyboard.
</figcaption>
</figure>
```

```
<section>
  <!--defines theme-->
  <h2>Top Sports league in
  America</h2>
  <!--writes independent content
  relating to that theme-->
  <article>
    <p>One of the top sports league
    is the nba.</p>
  </article>
</section>
```

```
<article>
  <!--Main Content-->
</article>
<aside>
  <!--Additional information-->
</aside>
```