

420-KB2-LG, Conception d'interfaces Web

Préparé par Saliha Yacoub

Les éléments structurants

Plan de la séance:

- Les éléments structurants ou sémantiques
 - Header
 - Footer
 - Section
 - Nav
 - Section
 - Article
 - Aside
 - Main
- display:table
- Exemple
- Début du Tp2

Retour sur la
dernière
séance

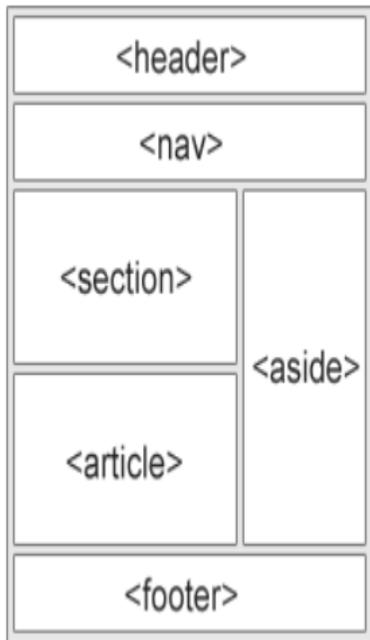
Point de vue des
étudiants

Point de vue de
l'enseignant

Éléments structurants ou éléments sémantiques

- On entend par élément sémantique un élément qui décrit clairement le sens de son contenu.
- Par exemple, tout le monde sait ce qu'est un titre (h1 à h6), un paragraphe (p) ou un tableau (table).
- À l'opposé, des éléments comme div et span ne disent rien sur le sens de leur contenu. Ce ne sont que des conteneurs.
- Les éléments que nous allons présenter ici ont été introduits avec HTML5 et sont la fondation d'une page Web moderne.

Les unités de mesures



- article
- aside (barre latérale)
- details (informations secondaires)
- figcaption (légende pour figure)
- figure
- footer
- header
- main (contenu principal)
- nav
- section
- summary (légende pour details)

Il s'agit du genre de page Web que l'on voit le plus souvent, avec comme éléments fondamentaux :

- un en-tête;
- une barre avec des éléments de navigation (boutons, hyperliens);
- une barre latérale qui peut être positionnée à gauche ou à droite;
- des sections qui contiennent des articles ou des articles découpés en sections;
- un pied de page.

Éléments structurants ou éléments sémantiques

L'élément header

- Cet élément représente l'en-tête d'un document, d'une section ou d'un article.
- Il doit être utilisé comme conteneur pour tout matériel d'introduction.
- On peut retrouver plusieurs éléments header à l'intérieur d'un même document.
- Il peut contenir un logo, une barre de navigation etc...

Éléments structurants ou éléments sémantiques

L'élément footer

- Cet élément représente le pied de page d'un document, d'une section ou d'un article.
- Il doit être utilisé comme conteneur pour toute information relative à son parent et qui ne figure pas dans l'en-tête.
- On y retrouve typiquement le nom de l'auteur, des informations de copyright, des informations de contact, des liens, etc.
- On peut retrouver plusieurs éléments footer à l'intérieur d'un même document.

Éléments structurants ou éléments sémantiques

L'élément nav

- Cet élément regroupe un ensemble d'hyperliens de navigation.
- Tous les hyperliens d'une page n'ont pas à se retrouver nécessairement dans l'élément nav, mais seulement les plus importants.

```
<nav>  
  Choisissez votre destination :  
  <a href="">La Comté</a> |  
  <a href="">Isengard</a> |  
  <a href="">Ithilien</a> |  
  <a href="">Mordor</a>  
</nav>
```

Éléments structurants ou éléments sémantiques

L'élément aside

- Cet élément représente une barre latérale placée à gauche ou à droite.
- L'élément aside est généralement utilisé comme conteneur pour un menu, de la publicité, etc.
- Notons que le fait d'insérer un élément aside dans le document ne le dispose pas automatiquement à gauche ou à droite du contenu principal.
- Pour cela il est nécessaire d'utiliser certaines propriétés CSS

Éléments structurants ou éléments sémantiques

L'élément section

- Cet élément représente une section dans un document.
- Une section est considérée comme un regroupement d'informations sur un sujet donné, incluant typiquement un en-tête (élément header).
- Une page d'accueil peut par exemple être découpées en sections : une pour l'introduction, une pour le contenu, une pour les informations de contact, etc.

Éléments structurants ou éléments sémantiques

L'élément article

- Cet élément représente un contenu indépendant du reste du document.
- Il devrait être possible de lire un article indépendamment du reste de la page ou du site Web.
- Exemples d'utilisation de l'élément article :
 - publication dans un forum
 - publication sur un blog
 - annonce classée
 - article de journal

Éléments structurants ou éléments sémantiques

Un article dans une section? Une section dans un article?

- Maintenant il est légitime de se poser les questions suivantes :
 - peut-on séparer un article en sections?
 - est-ce qu'une section d'un document peut contenir des articles?
- La réponse aux deux questions est OUI.
- On retrouve même des sections dans des sections (des "sous-sections") et des articles dans des articles.
- L'important est de bien réfléchir avant de choisir la structure du document et d'être cohérent ensuite.

Éléments structurants ou éléments sémantiques

Les éléments figure et figcaption

Terminons cette présentation par deux nouveaux éléments.

- Le but d'un figcaption est d'ajouter des explications à une image, ce qu'on appelle généralement la légende.
- L'élément figure, lui, permet de regrouper l'image et la légende au sein d'un même élément.

```
<figure>
  
  <figcaption>
    Fallaout, le jeu le plus cool
  </figcaption>
</figure>
```

```
figure {
  background-color: white;
  border: solid black 1px;
  padding: 10px;
  width: fit-content;
  float: left;
}

figcaption {
  text-align: left;
  font-style: italic;
}
```

Éléments structurants ou éléments sémantiques

L'élément main

L'ensemble des éléments que nous venons de définir peuvent être contenu dans l'élément main. On peut y exclure les éléments footer et header.

https://www.w3schools.com/html/html5_semantic_elements.asp

Les grilles et les éléments structurants

Comment placer tous ces éléments ?

- Ce sont des éléments structurants, ce ne sont pas des éléments de mise en page.(comme un paragraphe)
- Pour une belle mise en page vous avez pour l'instant: les div.
- En utilisant les grilles CSS.

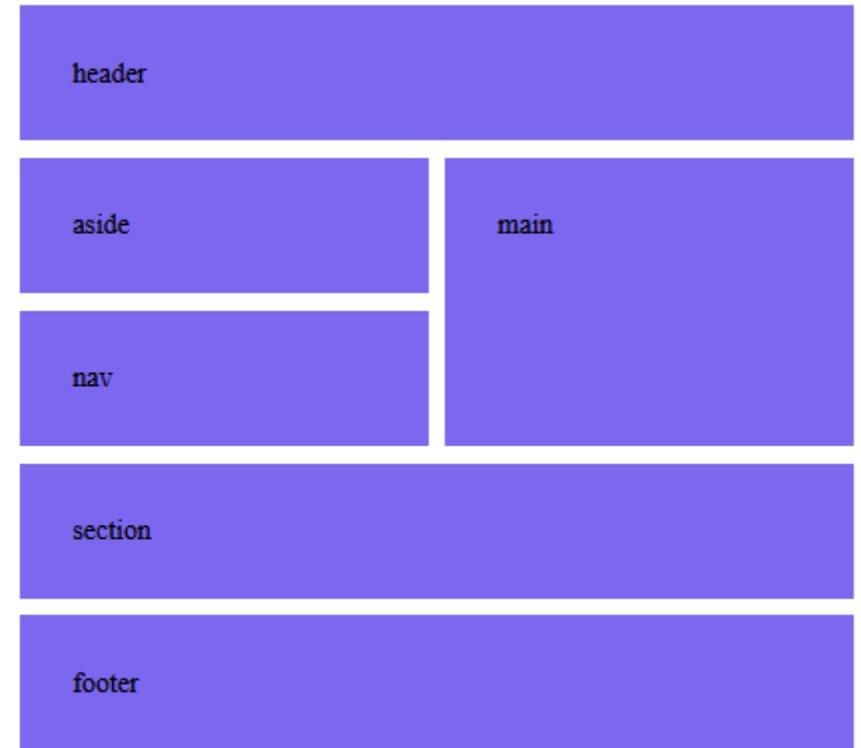
Les grilles et les éléments structurants

- La propriété `grid-area` d'une grille peut également être utilisée pour attribuer un nom à un élément de grille. Les éléments de grille nommés peuvent ensuite être référencés par la propriété `grid-template-areas` du conteneur de grille.
- Peuvent ensuite être référencés et placés directement dans la grille.
- Cette fois nous allons utiliser les éléments sémantiques appropriés et non plus des conteneurs généraux comme l'élément `div`
- Sans spécifier le nombre de colonnes et de rangées, ni la position exacte des éléments, ces derniers s'empilent simplement à la verticale .

Les grilles et les éléments structurants

- Imaginons que nous voulons obtenir la grille suivante:
- Nous avons un div principal, mais le reste des éléments sont structurants.
- Nous devons associer aux différents éléments structurants un grid-area de la façon suivante:

```
header {  
  grid-area: header;  
}  
main {  
  grid-area: main;  
}  
section {  
  grid-area: section;  
}  
Etc...
```



Les grilles et les éléments structurants

- Ensuite pour changer la position ou la superficie occupée par un élément se résume maintenant à "jouer" avec les zones du **grid-template-areas** dans le conteneur principal

```
grid-template-areas:  
  "header header header header"  
  "aside  aside  main  main"  
  "nav    nav    main  main"  
  "section section section section"  
  "footer footer footer footer";  
}
```

Les grilles et les éléments structurants

```
/*conteneur principal: la grille*/
#conteneur {
  display: grid;
  grid-template-columns: repeat(4, 1fr); /* 4
colonnes */
  grid-auto-rows: minmax(75px, auto); /*
autant de rangées qu'il faut */
  grid-gap: 10px;
  max-width: 480px;
  margin: auto;
  grid-template-areas:
    "header header header header"
    "aside aside main main"
    "nav nav main main"
    "section section section section"
    "footer footer footer footer";
}
#conteneur > * {
  background-color:mediumslateblue;
  padding: 30px;
}
```

```
/*les grid area*/
header {
  grid-area:
header;
}
main {
  grid-area: main;
}
section {
  grid-area:
section;
}
aside {
  grid-area: aside;
}
nav {
  grid-area: nav;
}
footer {
  grid-area:
footer;
}
```

```
<!--le code HTML-->
<div id="conteneur">
  <header>header</header>
  <main>main</main>
  <section>section</section>
  <aside>aside</aside>
  <nav>nav</nav>
  <footer>footer</footer>
</div>
```

[Cliquez ici pour le code CSS](#)

Affichage en colonnes, une façon de placer les éléments

Comment placer tous ces éléments ?

- Ce sont des éléments structurants, ce ne sont pas des éléments de mise en page.(comme un paragraphe)
- Pour une belle mise en page vous avez pour l'instant: les div.
- Mais les div sont parfois difficile à positionner → bientôt nous verrons les grilles CSS qui sont un excellent moyen pour la mise en page. display grid.
- La propriété display:table d'un élément (d'un div par exemple) pourrait également aider à positionner des éléments structurant sans trop de problème.

Affichage en colonnes

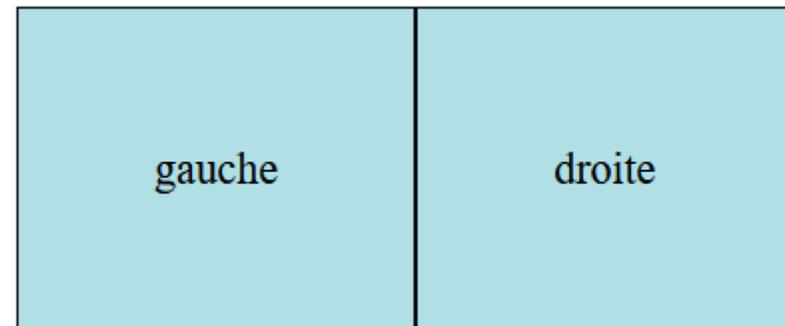
La propriété css display:table

- La propriété display:table permet d'afficher un élément HTML comme s'il s'agissait d'un tableau.
- On parle ici de la "propriété" display:table, tout en sachant que pour être rigoureux on devrait dire la valeur "table" de la propriété "display".
- display:table est surtout utilisé sur des éléments div.
- Elle peut également être utilisée pour les éléments structurants en particulier main lorsque vous voulez afficher les section côte à côte
- Deux autres valeurs possibles de la propriété display permettent de représenter les rangées (**display:table-row**) et les cellules (**display:table-cell**).
- Lorsque votre tableau a juste une ligne, le display: table-row peut être omis (enlevé)

Affichage en colonnes

```
.tableau {
  display: table;
}
.rangee {
  display: table-row;
}
.cellule {
  display: table-cell;
  padding: 50px;
  background-color: powderblue;
  border: solid black 1px;
}
```

```
<body>
  <div class="tableau">
    <div class="rangee">
      <div class="cellule">gauche</div>
      <div class="cellule">droite</div>
    </div>
  </div>
</body>
```

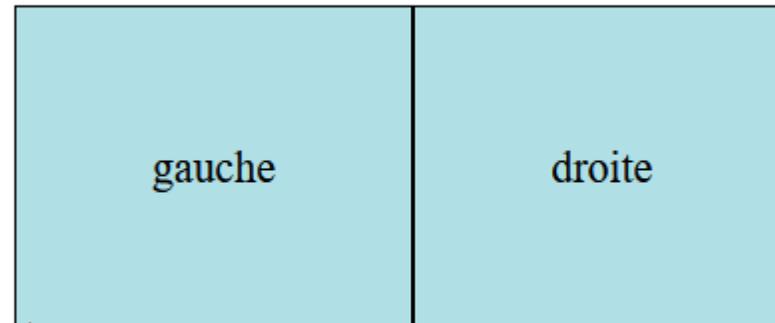


Affichage en colonnes

Remarquez que nous n'avons pas de ligne pour ce tableau ce qui donne le même résultat

```
.tableau {  
  display: table;  
}  
.cellule {  
  display: table-cell;  
  padding: 50px;  
  background-color: powderblue;  
  border: solid black 1px;  
}
```

```
<body>  
  <div class="tableau">  
    <div class="cellule">gauche</div>  
    <div class="cellule">droite</div>  
  </div>  
</body>
```



Affichage en colonnes

- Nous avons vu comment il est possible de créer un tableau sans faire appel à l'élément table.
- Pour un logiciel qui manipule un document HTML, rien ne garantit que le premier élément rencontré soit un en-tête ou que le n-ème élément est le contenu principal.
- Il est hautement préférable d'utiliser des éléments structurants comme header, nav, main, section, etc.
- L'utilisation de la propriété `display:table` nous permettra alors quand même de créer une mise en page en colonnes, tout en préservant le sens de chacun des éléments
- Plus loin nous verrons la propriété `display:grid` d'un élément.

```
main{
  display:table;
  width: 98%;
  height: 100%;
}
article{
  display: table-cell;
  width: 70%;
  height: 100%;
}

aside {
  display: table-cell;
  width: 25%;
  height: 100%;
}
```

Questions



CONCLUSION



QUESTIONS ??

Sources

- <https://www.w3schools.com/html/>
- <https://prog101.com/cours/kb2/>