

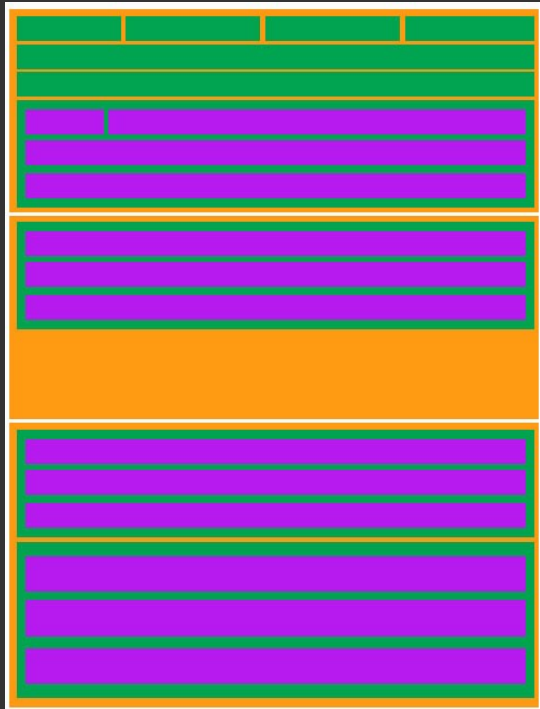
Intégration Web

MMI 1 – TP#8 S1

Danielo **JEAN-LOUIS**
Michele **LINARDI**

Disposition des balises

- Par défaut, les balises ("inline" ou "block") se suivent et s'empilent respectivement



◀ Ci-contre, voici un exemple de mise en page que vous pouvez réaliser avec ce qu'on a vu en cours

Jusqu'à présent, notre mise en page n'a été composée que de colonnes, mais il est possible de faire des mises en page plus complexes

Notamment juxtaposer des balises

Mise en page - Historique

- Au début, les développeurs utilisaient des tableaux (<table>) pour faire la mise en page
 - **Ne jamais, jamais utiliser ceci !**
 - A utiliser uniquement :
 - Pour afficher des données tabulaires
 - Faire des e-mails

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/float>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/table>
- <https://www.alsacreations.com/article/lire/1209-display-inline-block.html>
- <https://putaindecode.io/podcasts/s02e04-css-layout> – podcast en français

Mise en page - Historique

- Ensuite la propriété "float"
 - **A éviter. Crée plein d'effets de bord**
 - **Adapté pour certaines mise en page**

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/float>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/table>
- <https://www.alsacreations.com/article/lire/1209-display-inline-block.html>
- <https://putaindecode.io/podcasts/s02e04-css-layout> – podcast en français

Mise en page - Historique

- Puis la valeur "inline-block" pour la propriété "display"
 - Très limité dans l'usage
- Et puis le CSS3 apporta **flexbox**
 - Et **grid** aussi, mais on va voir **flexbox** uniquement pour le moment

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/float>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/table>
- <https://www.alsacreations.com/article/lire/1209-display-inline-block.html>
- <https://putaindecode.io/podcasts/s02e04-css-layout> – podcast en français

Mise en page - Flexbox

- Propriété apparue avec CSS3
- **Compatible avec tous les navigateurs modernes**
 - **Donc pas d'Internet Explorer**
- Permet des mises en page complexes **sur une seule dimension** (colonne ou ligne)

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout
- <https://css-tricks.com/snippets/css/a-guide-to-flexbox/> - anglais
- <https://putaindecode.io/podcasts/s02e04-css-layout> – podcast en français
- <https://philipwalton.github.io/solved-by-flexbox/> - anglais

Mise en page - Flexbox

- Flexbox resolvera 90 % de vos problèmes de mise en page web
 - **Pensez toujours à flexbox**

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout
- <https://css-tricks.com/snippets/css/a-guide-to-flexbox/> - anglais
- <https://putaindecode.io/podcasts/s02e04-css-layout> – podcast en français
- <https://philipwalton.github.io/solved-by-flexbox/> - anglais

Exemples de mise en page non-réalisables sans flexbox

<https://philipwalton.github.io/solved-by-flexbox/>

Flexbox – Pré-requis

- A besoin impérativement d'un parent flex
 - La balise parente doit avoir la valeur "flex" pour la propriété "display"

display: flex;

```
<body>
  <div class="est-flexbox"> <!-- cette balise possède la déclaration css "display: flex" -->
    <select name="" id="">[...]</select>
    <div class="enfant">[...]</div>
    <ul></ul>
  </div>
</body>
```

Les enfants sont maintenant des éléments flexbox

Sources :

- <https://developer.mozilla.org/fr/docs/Web/CSS/flex>

Flexbox - Possibilités

- Faire une mise en page en colonne ou en ligne
 - Propriété "flex-direction"
- Gestion des espaces et du positionnement dans l'espace

Sources :

- <https://developer.mozilla.org/fr/docs/Web/CSS/flex>

Flexbox - Possibilités

- Changement de l'ordre des éléments
- Les enfants d'une balise flexbox peuvent eux-même être flexbox
 - Et ainsi de suite
 - Tout dépendra de vos designs

Sources :

- <https://developer.mozilla.org/fr/docs/Web/CSS/flex>
- <https://openclassrooms.com/fr/courses/1603881-apprenez-a-creer-votre-site-web-avec-html5-et-css3/3298561-faites-votre-mise-en-page-avec-flexbox>

Consultez ces exemples pour mieux comprendre

- <https://la-cascade.io/flexbox-guide-complet/>
- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout/Typical_Use_Cases_of_Flexbox
- <https://css-tricks.com/snippets/css/a-guide-to-flexbox/> - anglais
- <https://flexbox.help/> - anglais

Jeu d'apprentissage sur flexbox :
<https://flexboxfroggy.com/#fr>

Essayez-le

Pratiquons ! - Découvrons flexbox

Appliquez ce que vous venez d'apprendre en ajoutant de nouvelles zones à votre projet avec ces nouvelles notions

Pour aller plus loin

Pré-requis :

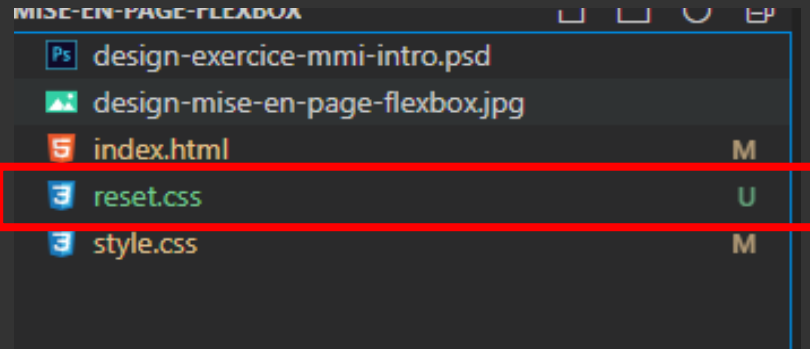
- Avoir la ressource **ressources/flexbox**

<https://download-directory.github.io/?url=https%3A%2F%2Fgithub.com%2FDanYellow%2Fcours%2Ftree%2Fmain%2Fintegration-web-s1%2Ftravaux-pratiques%2Fnumero-8%2Fresources>

Correction incluse

Pratiquons ! - Découvrons flexbox, mise en page

Contenu de la ressource



Reset css

- S'oppose à la philosophie du normalize.css
 - Les balises conservent leur style par défaut mais il est uniforme entre les navigateurs
- **Premier fichier CSS à charger dans la liste des <link>**

Source(s) :

- <https://www.alsacreations.com/astuce/lire/36-reset-css.html>
- <https://meyerweb.com/eric/tools/css/reset/> - anglais
- <https://stackoverflow.com/questions/6887336/what-is-the-difference-between-normalize-css-and-reset-css> - anglais
- <https://necolas.github.io/normalize.css/>

Reset css

- Permet de réinitialiser les styles par défaut des navigateurs, toutes les balises ont un style "nul"
 - Par exemple les balises h1 et p ont la même taille
- Attention tous les resets ne valent pas
 - Eric Meyer Reset le plus populaire

Source(s) :

- <https://www.alsacreations.com/astuce/lire/36-reset-css.html>
- <https://meyerweb.com/eric/tools/css/reset/> - anglais
- <https://stackoverflow.com/questions/6887336/what-is-the-difference-between-normalize-css-and-reset-css> - anglais
- <https://necolas.github.io/normalize.css/>

Mise en page – CSS Grid

- Bien plus puissant que flexbox : grid
 - Permet de gérer les éléments sur deux axes : horizontal et vertical
 - Il existe des générateurs de code CSS Grid en ligne
 - <https://grid.layoutit.com/>

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/flex>
- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Grid_Layout
- <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>
- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout/Concepts_de_base_flexbox

Mise en page – CSS Grid

- ...Et même un jeu pour apprendre :
 - <https://codepip.com/games/grid-garden/#fr>
 - Nécessite un compte. Possibilité de s'inscrire avec une adresse yopmail
 - Adresse jetable

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/flex>
- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Grid_Layout
- <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>
- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout/Concepts_de_base_flexbox

Flexbox – En résumé

- Flexbox sera un atout de taille pour réaliser vos mise en page web, **il faut toujours penser à lui**

Questions ?