

Intégration Web

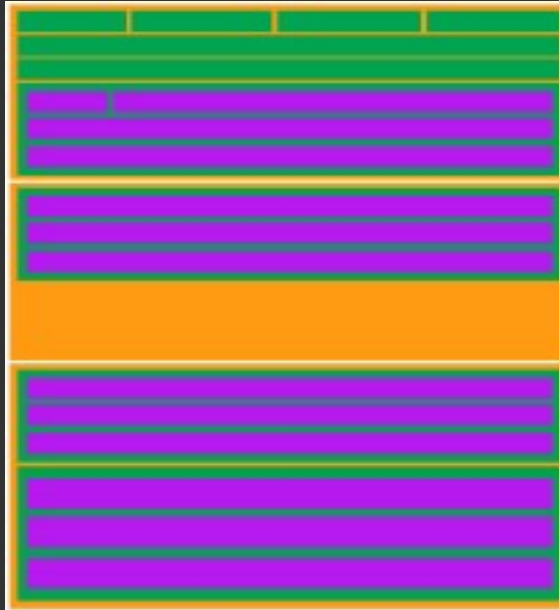
MMI 1 – TP#8 S1



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

Disposition des balises

- Par défaut, les balises ("inline" ou "block") se suivent et s'empilent respectivement
 - On dit qu'elles sont dans le flux



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 tout
en les gardant dans le flux**

Mise en page - Historique

- Au début, les développeurs utilisaient des tableaux (<table>) pour faire la mise en page
 - **Ne jamais, jamais utiliser ceci !**
- Ensuite la propriété "float"
 - **A éviter. Crée plein d'effets de bord**
 - **Adapté pour certaines mise en page**

Sources :

- <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"
- Et puis le CSS3 apporta **flexbox**
 - Et **grid** aussi, mais on va voir **flexbox** uniquement pour le moment

Sources :

- <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
- Permet des mises en page complexes **sur une seule dimension** (colonne ou ligne)

Sources :

- 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

Sources :

- 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

- A besoin 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>
- <https://openclassrooms.com/fr/courses/1603881-apprenez-a-creeer-votre-site-web-avec-html5-et-css3/3298561-faites-votre-mise-en-page-avec-flexbox>
- Liste bugs flexbox : <https://github.com/philipwalton/flexbugs> - anglais

Mise en page - Flexbox

- Possibilité de faire une mise en page en colonne ou en ligne (propriété "flex-direction")
- Gestion des espaces et du positionnement dans l'espace
- Changement de l'ordre des éléments
- Les enfants d'une balise flexbox peuvent eux-même être flexbox

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>

Mise en page - Flexbox

- La propriété "flex-direction" définit l'axe principal de l'alignement
 - propriété "justify-content" définit le comportement de l'axe primaire
 - propriété "align-items" définit le comportement de l'axe secondaire

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>

Mise en page - Flexbox

- En ajoutant "-reverse", il est possible d'inverser l'ordre des enfants d'un conteneur flexbox
 - Exemple "flex-direction : row-reverse"
- En mettant la déclaration "margin:auto" aux enfants, ils se centrent tous seuls verticalement et horizontalement
- Un enfant d'un élément flexbox peut être lui-même flexbox

Sources :

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

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
- Testeur de flexbox : <https://flexbox.help/>
 - Essayez-le

Pratiquons ! - Découvrons flexbox

Pré-requis :

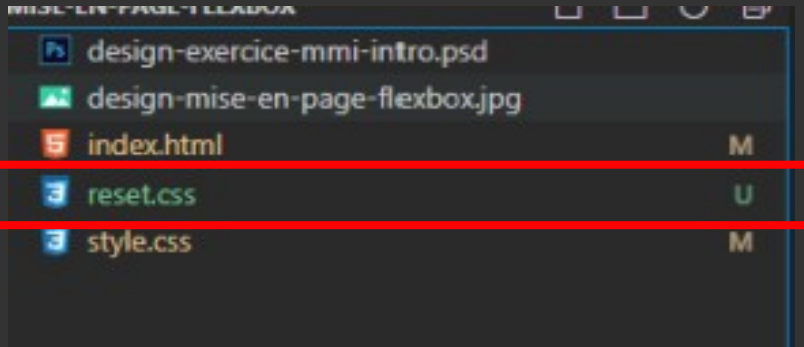
- Avoir la ressource **ressources/site-flexbox**

A télécharger sur ENT dans la partie "Ressource TP" > "TP 8"

Connectez-vous avec votre compte Teams/Office 365 pour télécharger la ressource

Pratiquons ! - Découvrons flexbox, mise en page

Contenu de la ressource



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

Sources :

- <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

- 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**

Sources :

- <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/>

Pratiquons ! - Découvrons flexbox, mise en page

[Accueil](#)

[Galerie](#)

[Contact](#)

- Ajout de la valeur "flex" pour la propriété "display" de la navigation
- Utilisation de la déclaration "margin-left : auto" pour ferrer à droite l'élément de navigation "contact"

Pratiquons ! - Découvrons flexbox, mise en page



- Utilisation des propriétés :
 - padding
 - 16 px en haut et en bas / 30 px à gauche et à droite
 - border
 - 1px d'épaisseur
 - style : solid
 - couleur : 5d5d5d

Pratiquons ! - Découvrons flexbox, mise en page



- Par défaut, les enfants des éléments flex occupent l'espace de leur contenu. Le navigateur s'arrange pour tout faire rentrer
- Mettre une taille fixe ne fonctionne pas comme attendu

Sources :

- <https://www.alsacreations.com/tuto/lire/1493-css3-flexbox-layout-module.html>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-shrink>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-grow>
- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout/Concepts_de_base_flexbox

Pratiquons ! - Découvrons flexbox - flex-basis



- La propriété "flex-basis" permet de définir la taille d'un élément avant que l'espace soit distribué
- "flex-basis" écrase la valeur de la propriété "width"
- Accepte une longueur comme valeur
- Représente la troisième valeur de la propriété "flex"

Sources :

- <https://www.alsacreations.com/tuto/lire/1493-css3-flexbox-layout-module.html>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-shrink>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-grow>
- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout/Concepts_de_base_flexbox

Pratiquons ! - Découvrons flexbox - flex

- Propriété à appliquer sur les enfants direct d'éléments flex
- Flex propriété raccourcie contenant :
 - flex-grow : capacité à occuper l'espace restant
 - flex-shrink : rétrécissement d'un élément si nécessaire
 - flex-basis : taille de l'élément avant répartition de l'espace

Sources :

- <https://www.alsacreations.com/tuto/lire/1493-css3-flexbox-layout-module.html>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-shrink>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-grow>
- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout/Concepts_de_base_flexbox

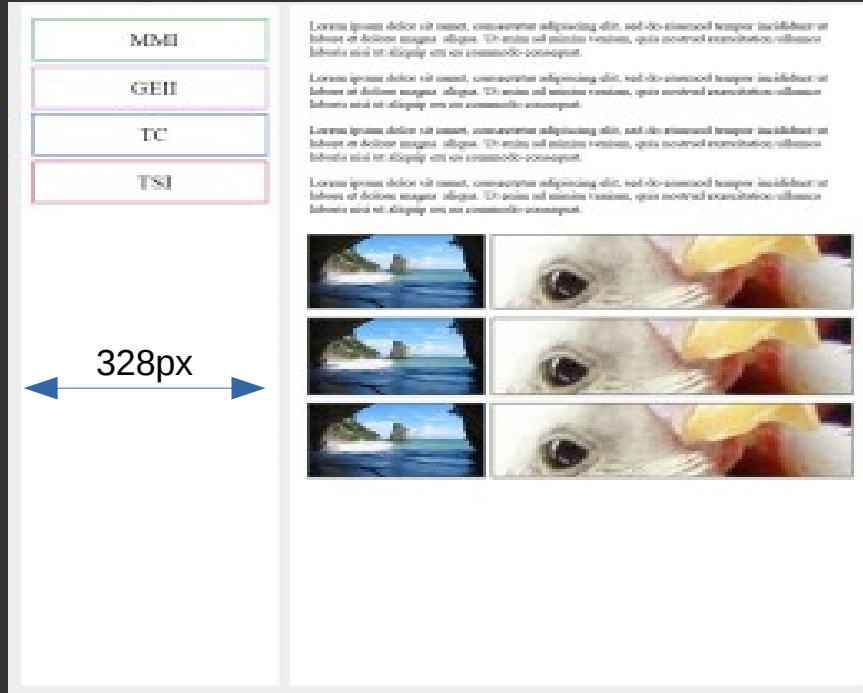
Pratiquons ! - Découvrons flexbox - flex

- "flex-grow" et "flex-shrink" acceptent un nombre décimal positif (si valeur incorrecte, valeur = 1)
- "flex-shrink: 0" indique que ma balise ne doit pas être rétrécie

Sources :

- <https://www.alsacreations.com/tuto/lire/1493-css3-flexbox-layout-module.html>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-shrink>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-grow>
- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout/Concepts_de_base_flexbox

Pratiquons ! - Découvrons flexbox, mise en page



- Avoir la navigation de gauche avec une largeur de 328px

```
.liste-cursus {  
  flex-basis: 328px;  
  flex-shrink: 0;  
  flex-grow: 0;  
}
```

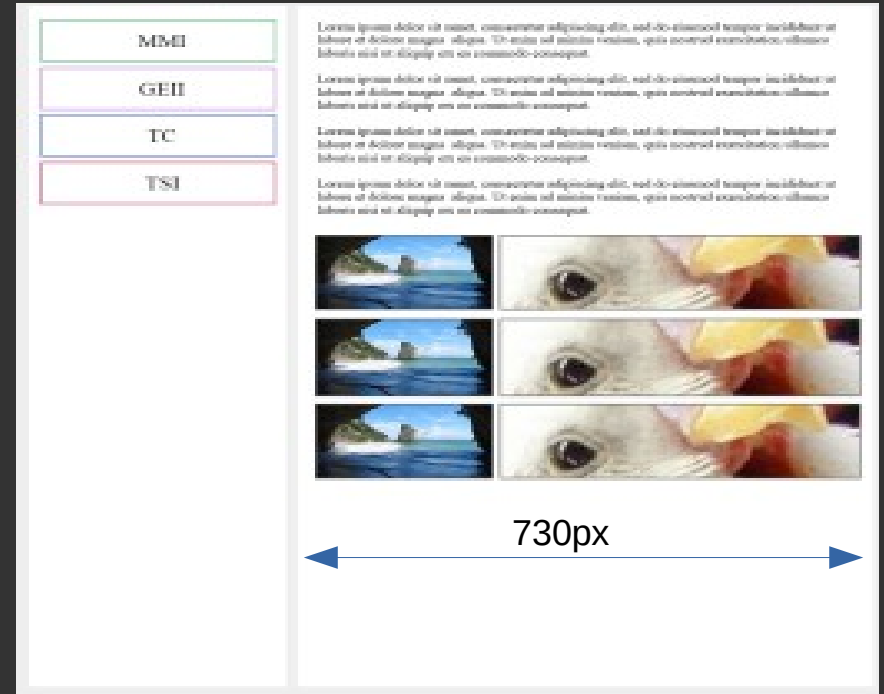
Sources :

- <https://www.alsacreations.com/tuto/lire/1493-css3-flexbox-layout-module.html>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-shrink>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-grow>
- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout/Concepts_de_base_flexbox

Pratiquons ! - Découvrons flexbox, mise en page

- Et la partie de droite avec une largeur de 730 px

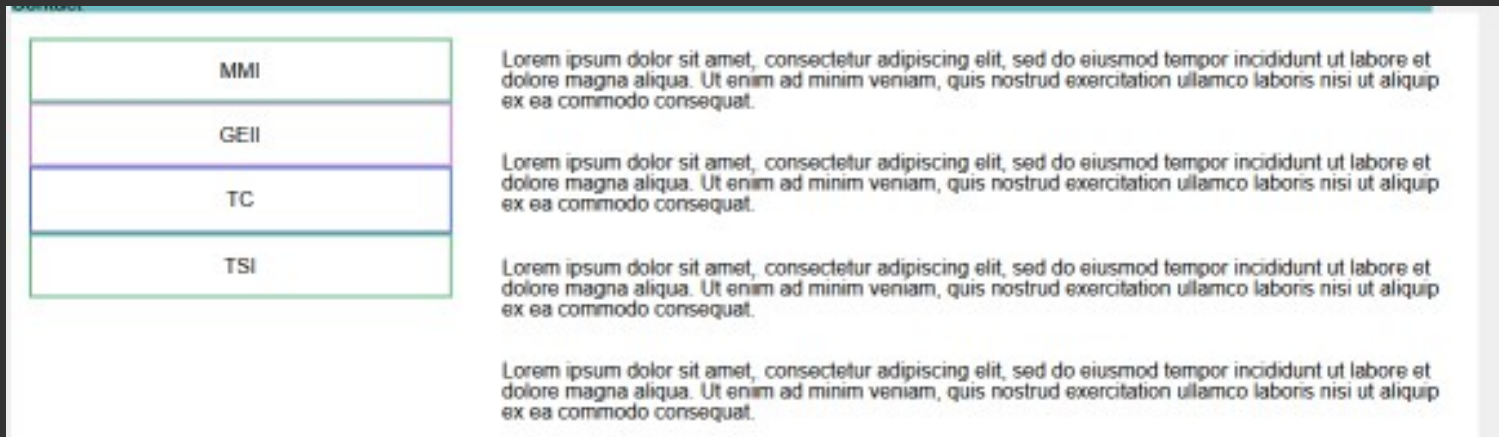
```
.description {  
    flex: 0 0 730px;  
}
```



Sources :

- <https://www.alsacreations.com/tuto/lire/1493-css3-flexbox-layout-module.html>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-shrink>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-grow>
- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout/Concepts_de_base_flexbox

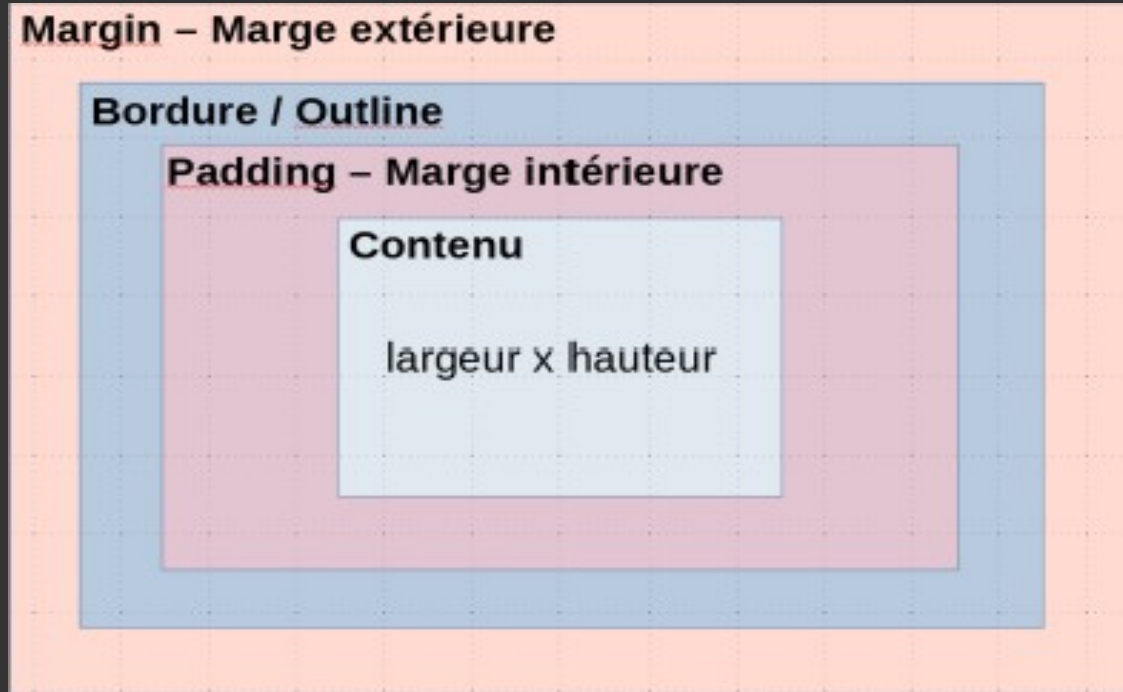
Pratiquons ! - Découvrons flexbox, mise en page



La mise en page ne ressemble pas vraiment à notre jpg malgré le bon dimensionnement des deux parties

Calcul de dimension de boite

Rappel du modèle de boite

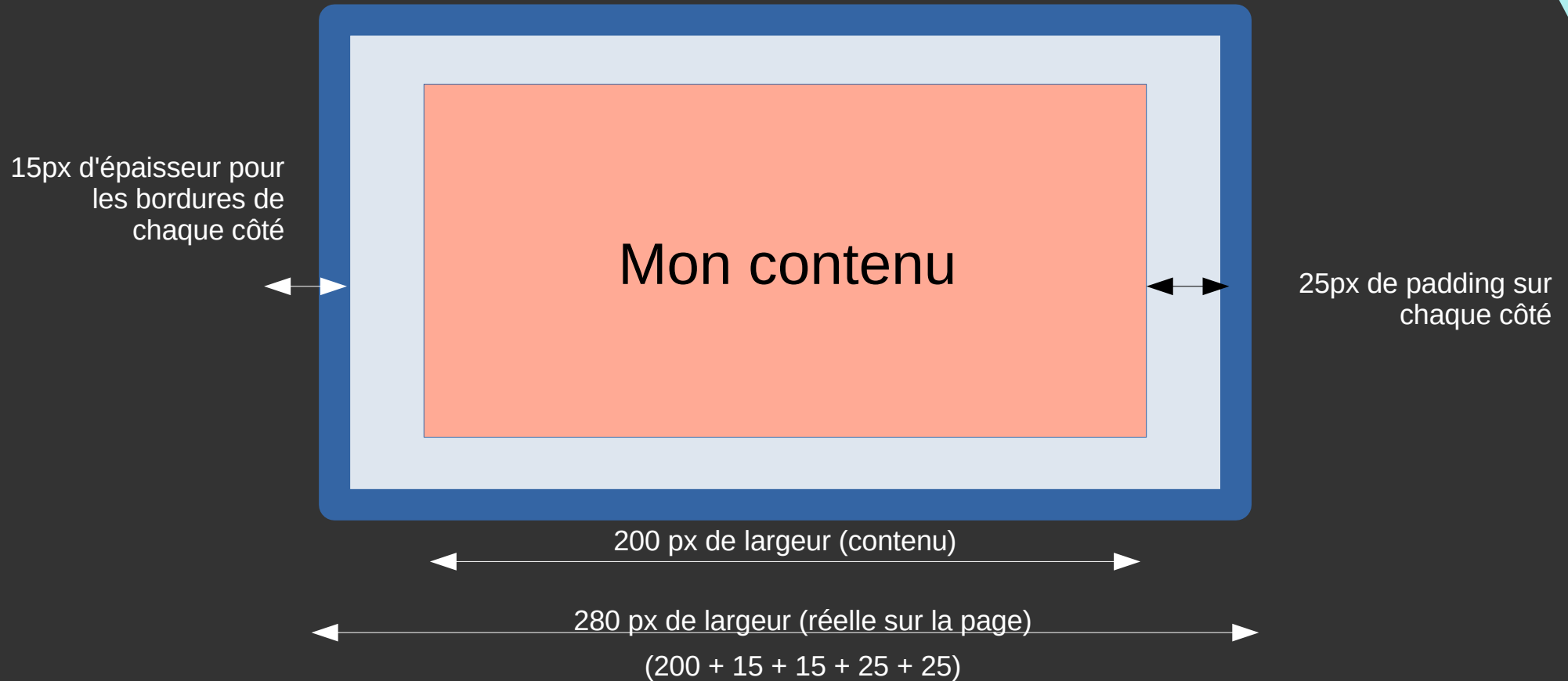


Calcul de dimension de boite

- Trois composantes du modèle de boite définissent la taille d'une balise :
 - contenu (width / height)
 - padding
 - border
- Admettons que j'écrive le code suivant :

```
.ma-balise {  
  width: 200px;  
  border: 15px solid blue;  
  padding: 25px;  
}
```

Calcul de dimension de boite



Bien que la "width" soit de 200px, l'ajout de "padding" et de "border" ne reflète plus les 200px de largeur

Il faut donc prendre en compte vos éventuelles border et margin quand vous définissez la "width" et la "height" d'une balise, mais...

Propriété box-sizing - Calcul de dimension de boite

- Permet de changer la façon dont est calculé les dimensions d'une boite
 - Moins de prise de tête
- Deux valeurs possibles
 - content-box : valeur par défaut
 - border-box : les propriétés "width" et "height" incluent la marge intérieure et la bordure

Sources :

- <http://fr.learnlayout.com/box-sizing.html>

Propriété box-sizing - Calcul de dimension de boîte



Propriété box-sizing - Calcul de dimension de boite

```
*, *::before, *::after {  
  box-sizing: border-box;  
}
```

Seul cas où utiliser le sélecteur `*` est fortement encouragé

Sources :

- <http://fr.learnlayout.com/box-sizing.html>

Pratiquons ! - Découvrons flexbox, mise en page

- Rajoutons la déclaration suivante à notre fichier css (au début du fichier)
- `*, *::before, *::after {`
 `box-sizing: border-box;`
}

Pratiquons ! - Découvrons flexbox, mise en page



Seul cas où utiliser le sélecteur * est légitime

**Ça avance, il nous faut maintenant les
espaces**

Pour rappel la propriété "justify-content" permet, sur un parent flex, de définir le comportement de ses enfants sur l'axe principal (ici x)

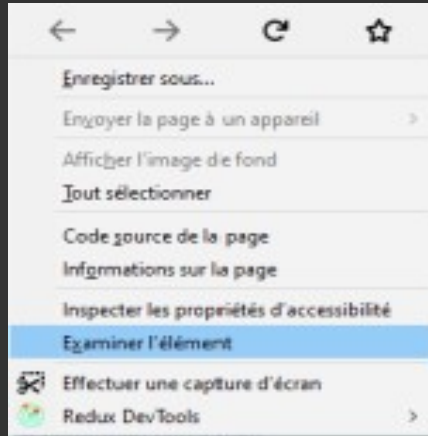
Testons la propriété... sans toucher à VS Code

Console du navigateur

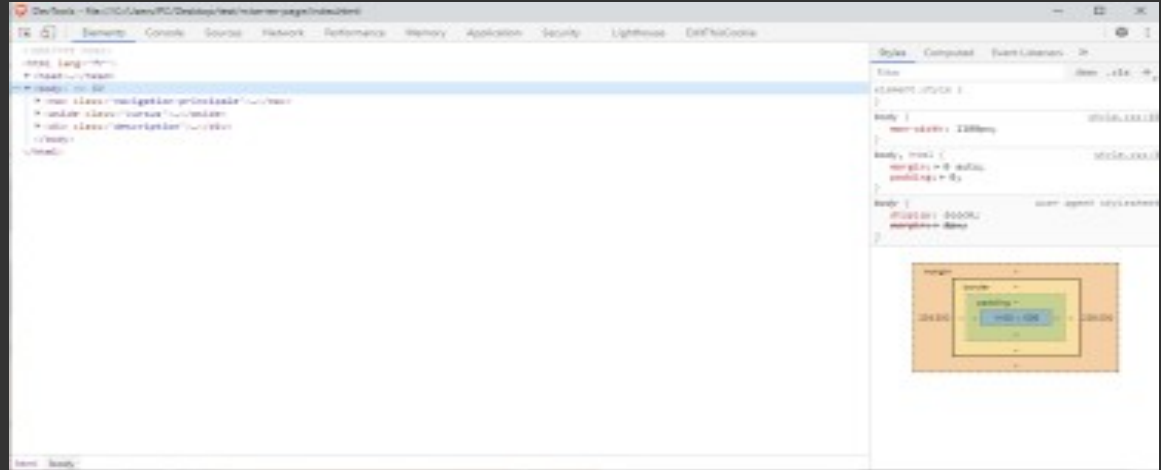
- Outil natif des navigateurs modernes
 - Même sous mobile avec un ordinateur
- Permet d'effectuer rapidement des modifications / tests sur une page web
- Les changements sont locaux
- **Indispensable pour debugger son site**
- Permet "d'émuler" des appareils mobiles

Console du navigateur

- Accessible via la touche F12 (Win) ou clic droit > Inspecter / examiner l'élément n'importe où sur la page



Menu contextuel de
Firefox

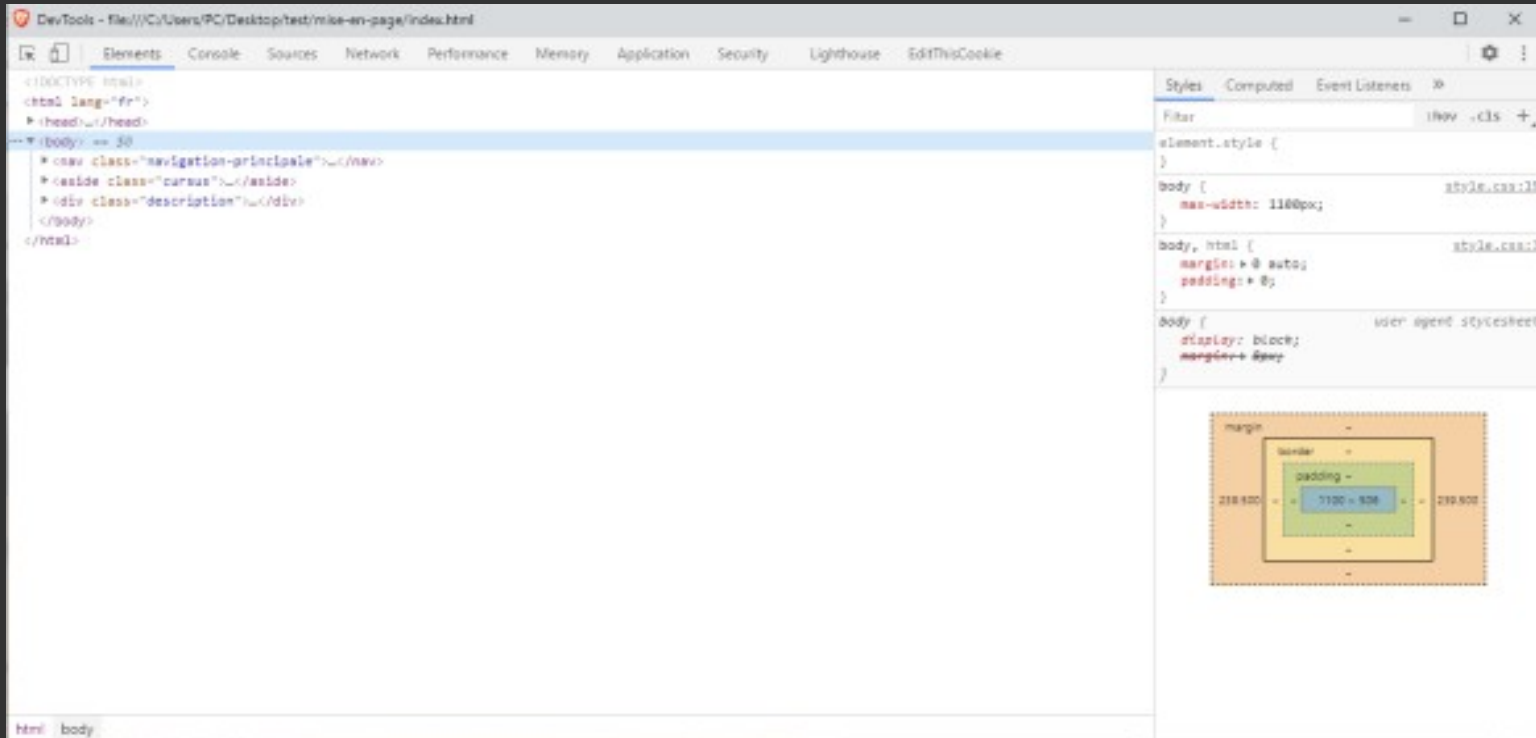


console du navigateur Brave

Sources :

- https://developer.mozilla.org/en-US/docs/Tools/Web_Console/Keyboard_shortcuts
- <https://developers.google.com/web/tools/chrome-devtools/shortcuts>

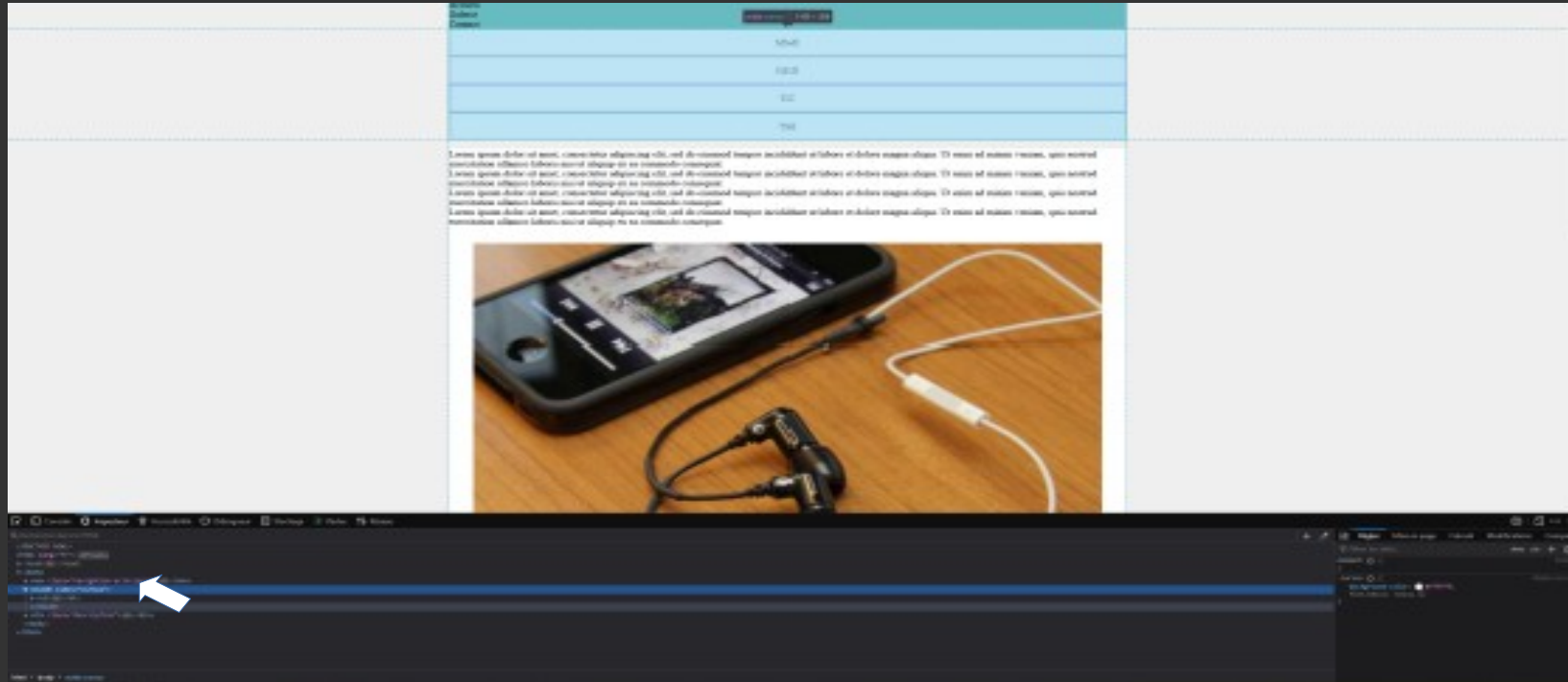
Console du navigateur



Sources :

- https://developer.mozilla.org/en-US/docs/Tools/Web_Console/Keyboard_shortcuts
- <https://developers.google.com/web/tools/chrome-devtools/shortcuts>

Console du navigateur – Onglet inspecteur



L'élément sélectionné dans l'onglet inspecteur / element est surligné sur la page

Sources :

- https://developer.mozilla.org/en-US/docs/Tools/Web_Console/Keyboard_shortcuts
- <https://developers.google.com/web/tools/chrome-devtools/shortcuts>

Console du navigateur – Onglet élément

```
> <p>_</p>
> <p>_</p>
  <p>_generates a 150 pixel square dummy image:~</p>
> <p>_</p>
> <p>_</p>
▼ <h2>
  <span id="How_To_Set_Image_Formats">How To Set Image Formats</span> == $0
  </h2>
> <p>_</p>
> <ul>_</ul>
```

Filter :hov .cls + [v]

Inherited from h2

```
h2 {
  autoptimize_9388 css:15
  font-size: 30px;
}
h1, h2, autoptimize_9388 css:15
h3, h4,
h5, h6 {
  clear: both;
  font-family: source sans
```

Déclarations CSS de l'élément
sélectionné

Sources :

- https://developer.mozilla.org/en-US/docs/Tools/Web_Console/Keyboard_shortcuts
- <https://developers.google.com/web/tools/chrome-devtools/shortcuts>

Pratiquons ! - Console web

Pré-requis :

- Avoir la ressource **ressources/site-flexbox**

A télécharger ici : <https://cours-prod.cyu.fr/mod/url/view.php?id=198104>

Connectez-vous avec votre compte Teams/Office 365 pour télécharger la ressource

- Chercher le conteneur principal de votre site
- Ajouter la propriété "justify-content" et jouer sur les valeurs possibles

Pratiquons ! - Console web

- Ajouter la déclaration
`"justify-content: space-around;"`
sur votre conteneur principal

Pratiquons ! - Console web



The screenshot displays a web application interface. On the left is a sidebar menu with four items: MMI, GEII, TC, and TSI. The main content area on the right contains four paragraphs of Lorem Ipsum text. At the bottom of the main content area, there is a large black rectangle with the text '600 x 480' in white, and a light blue rectangle below it, also containing the text '600 x 480' in black.

MMI
GEII
TC
TSI

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

600 x 480

600 x 480

Et voilà, notre conteneur principal ressemble beaucoup plus à la cible

Pratiquons ! - Découvrons flexbox, mise en page

- Toujours penser la segmentation de vos balises en blocs qui vont ensemble



bloc "galerie"

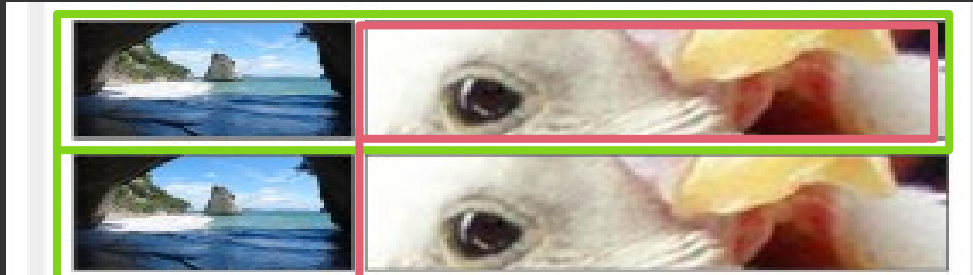
bloc "global" .description

Sources :

- <https://www.alsacreations.com/tuto/lire/1493-css3-flexbox-layout-module.html>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-shrink>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-grow>

Pourcentages – Premiers pas vers le responsive design

- Utilisation de pourcentages pour gérer les tailles



largeur du parent

largeur de
l'élément cible

Pourcentages – Premiers pas vers le responsive design

- Utilisation du produit en croix

$$\text{longueur en \%} = \frac{\text{longueur de l'élément cible} \times 100}{\text{longueur du parent}}$$

Pourcentages – Premiers pas vers le responsive design

- Largeur des petites images : 224px (32.230211583%)
- Largeurs des grandes images: 462px (66.474882014%)

```
.image-complementaire:nth-child(odd) {  
  flex: 0 0 32.230211583%;  
}
```

On cible les un élément sur deux en commençant par l'élément impair

Sources :

- <https://css-tricks.com/examples/nth-child-tester/> - en
- <https://developer.mozilla.org/en-US/docs/Web/CSS/:nth-child>

Pourcentages – Premiers pas vers le responsive design

- Largeur des petites images : 224px (32.230211583%)
- Largeurs des grandes images: 462px (66.474882014%)

```
.image-complementaire:nth-child(even) {  
    flex: 0 0 66.474882014%;  
}
```

On cible les un élément sur deux en commençant par l'élément pair

Sources :

- <https://css-tricks.com/examples/nth-child-tester/> - en
- <https://developer.mozilla.org/en-US/docs/Web/CSS/:nth-child>

Pourcentages – Premiers pas vers le responsive design



Nos éléments ont bien la "bonne" taille, en revanche ils ne forment pas de colonnes comme sur le design



Pratiquons ! - Découvrons flexbox - flex-wrap

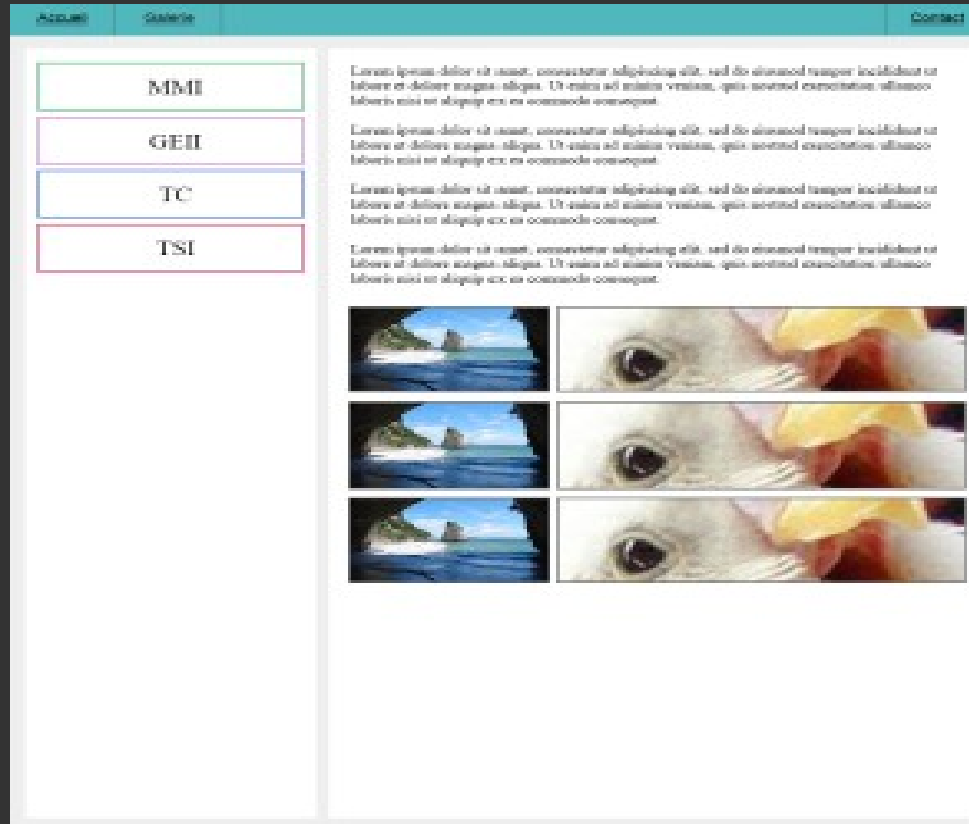


- La propriété flex-wrap indique si les enfants d'un élément flexbox doivent retourner à la ligne
- Possibilité de changer la direction de chaque ligne grâce à la valeur "wrap-reverse"

Sources :

- <https://www.alsacreations.com/tuto/lire/1493-css3-flexbox-layout-module.html>
- <https://developer.mozilla.org/fr/docs/Web/CSS/flex-wrap>
- https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout/Concepts_de_base_flexbox

Pratiquons ! - Découvrons flexbox



On devrait obtenir peu ou prou ce qu'on a en design

Mise en page – CSS Grid

- Flexbox possède une petite sœur bien plus puissante : "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/>

Sources :

- <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 ?