

Intégration Web

MMI 1 – TP#3 S1



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

Pratiquons ! - Modèle de boîte (Partie 1)

Pré-requis :

- Avoir la ressource ressources/boite

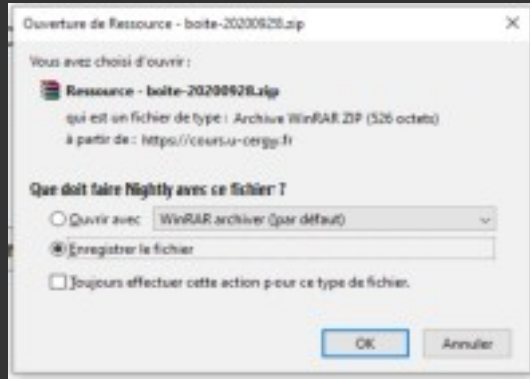
A télécharger ici :

<https://download-directory.github.io/?url=https://github.com/DanYellow/cours/tree/main/integration-web-s1/travaux-pratiques/numero-3/ressources>

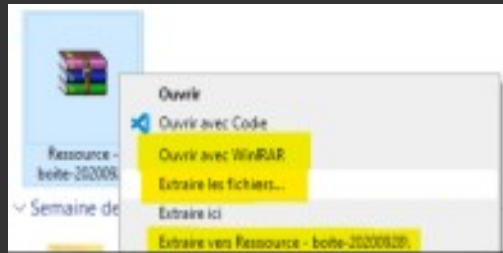
Sources :

- <https://code.visualstudio.com/Download>

Petit point sur les ressources de cours

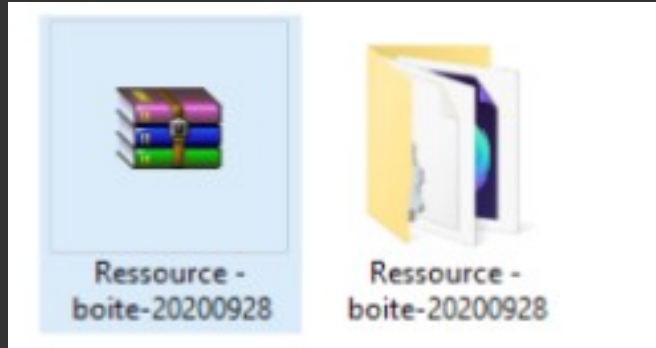


- Les ressources se présentent sous la forme d'une archive une fois téléchargées
- L'archive doit être impérativement extraite sinon vous ne serez pas capables de réaliser correctement les tp



Menu contextuel sur l'archive (clic droit). Choisir un des choix surligné.

Petit point sur les ressources de cours



- Une fois extrait, vous devez obtenir un dossier avec le contenu de l'archive
- Libre à vous de travailler dans le dossier OU en copier le contenu pour le coller ailleurs
- **En aucun vous devez travailler dans l'archive**
- Une fois fait, glissez le dossier dans VS Code

Pas très lisible, non ?

Point accessibilité – les couleurs

- **Beau n'est pas forcément égal à fonctionnel**
- Certaines personnes ne perçoivent pas tout le spectre colorimétrique (daltoniens par exemple)
- Pensez aux contrastes

Sources :

- <https://webaim.org/resources/contrastchecker/>
- <http://www.caf.fr/allocataires/vies-de-famille/vivre-avec-un-handicap/vos-droits/80-des-handicaps-sont-invisibles-le-saviez-vous>

Point accessibilité – les couleurs

- Comme la sémantique des balises html, les normes d'accessibilité ne sont que des recommandations
 - Mais un site accessible rend l'expérience bien meilleure et mémorable
- Les gens n'ont pas forcément des écrans super bien calibrés

Sources :

- <https://webaim.org/resources/contrastchecker/>
- <http://www.caf.fr/allocataires/vies-de-famille/vivre-avec-un-handicap/vos-droits/80-des-handicaps-sont-invisibles-le-saviez-vous>

Propriété "background-color"

- Permet de mettre une couleur d'arrière-plan sur (presque) toutes les balises html
- Accepte les valeurs suivantes :
 - Nom de couleur en anglais : red, pink, aliceblue...
 - Couleur au format rgb(a)
 - Couleur au format hexadécimal
 - Et autres

Sources :

- <https://developer.mozilla.org/fr/docs/Web/CSS/color>
- https://www.w3schools.com/css/css_colors.asp
- <https://developer.mozilla.org/fr/docs/Web/CSS/background-color>
- https://developer.mozilla.org/fr/docs/Web/CSS/Type_color

Couleur au format rgb(a)

- Représentation de la couleur sous la forme de quatre composantes :
 - Red (Rouge)
 - Green (Vert)
 - Blue (Bleu)
 - Alpha (Alpha) – Utilisation de la fonction rgba()
- Chaque valeur (sauf la composante alpha 0 – 1 valeur décimale) peut prendre une valeur entre 0 et 255 inclus

Sources :

- <https://developer.mozilla.org/fr/docs/Web/CSS/color>
- https://www.w3schools.com/css/css_colors.asp
- <https://developer.mozilla.org/fr/docs/Web/CSS/background-color>
- https://developer.mozilla.org/fr/docs/Web/CSS/Type_color

Couleur au format hexadécimal

- Représentation de la couleur sous la forme de quatre (ou trois) paires de valeurs hexadécimales précédées d'un croisillon (#):



Sources :

- <https://developer.mozilla.org/fr/docs/Web/CSS/color>
- https://www.w3schools.com/css/css_colors.asp
- <https://developer.mozilla.org/fr/docs/Web/CSS/background-color>
- https://developer.mozilla.org/fr/docs/Web/CSS/Type_color

Conversion hexadécimal vers décimal

- Chaque composante doit être multipliée par 16 élevé à la puissance de son index dans le chiffre
 - Note : On commence à compter à partir de 0 en programmation, le premier index est donc 0
- Puis on additionne le tout

index : 1 index : 0

$4 \times 16^1 (16) = 64$ $14 \times 16^0 (1) = 14$

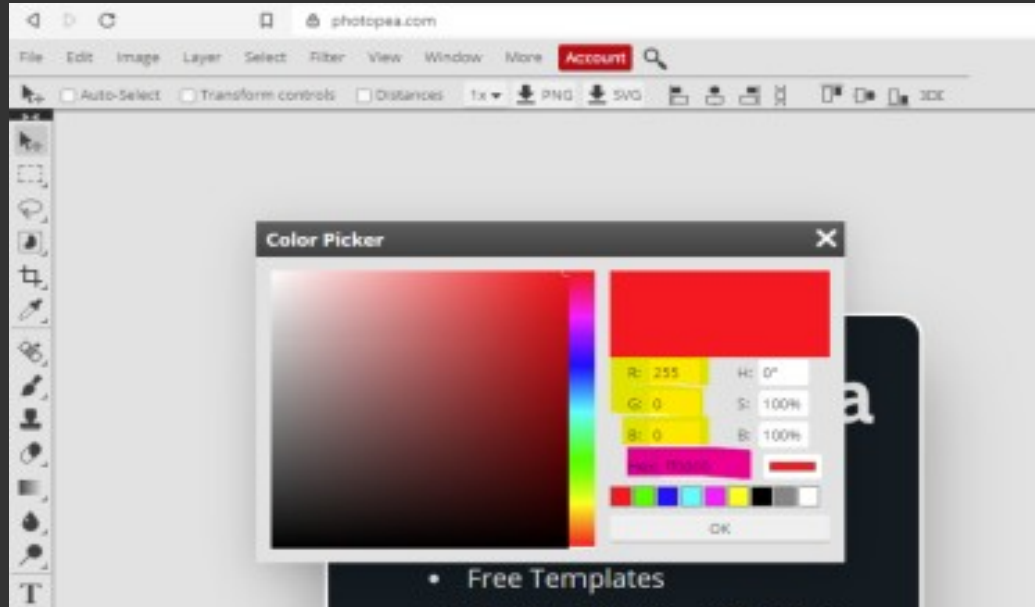
$64 + 14 = 78_{10}$ – base décimale

Sources :

- <https://www.rapidtables.com/convert/number/hex-to-decimal.html> - convertisseur

Hexadécimal ou rgb(a) ?

- Il n'y a pas un système moins bon que l'autre
- L'hexadécimal permet de récupérer plus facilement la couleur dans un logiciel



Sources :

- <https://developer.mozilla.org/fr/docs/Web/CSS/color>
- https://www.w3schools.com/css/css_colors.asp
- <https://developer.mozilla.org/fr/docs/Web/CSS/background-color>
- https://developer.mozilla.org/fr/docs/Web/CSS/Type_color

Propriété "background-color"

- La couleur d'arrière-plan peut être aussi définie dans la propriété "background"
 - Utilisation d'image (background-image)
 - Nécessite la fonction "url()" → par exemple :
url("../images/mon-image.jpg")
 - Utilisation de dégradés (background-image)
 - Permet l'utilisation de multiples background
 - Et plein d'autres propriétés du background sont raccourcies sous la propriété "background"

Sources :

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

Modèle de boîte - Introduction

- Chaque balise occupe un espace dans la page
- Espace sous forme de rectangle / carré
- *Schématisable* par le schéma suivant

Source :

- https://developer.mozilla.org/fr/docs/Web/CSS/Mod%C3%A8le_de_bo%C3%AEte_CSS

Modèle de boîte

Margin – Marge extérieure

Padding – Marge intérieure

Contenu

largeur x hauteur

Modèle de boîte - Contenu

- Par défaut, les balises de type "block" occupent 100 % de la largeur disponible
- Pour la hauteur, les balises occupent la hauteur de leurs contenus → Pas de contenu = hauteur nulle

Modèle de boîte - Contenu

- On peut modifier la taille de la boîte avec les propriétés :
 - width : largeur de boîte
 - height : hauteur de boîte

Modèle de boîte – Propriétés "height" et "width"

- Accepte les valeurs suivantes :
 - Longueur positive. Par exemple "42px"
 - Un pourcentage. Ex : 20 %
 - "auto" : le navigateur calcule la taille
 - Autres dont certaines sont encore expérimentales

Sources :

- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Le_modele_de_boite
- <https://developer.mozilla.org/fr/docs/Web/CSS/height>
- <https://developer.mozilla.org/fr/docs/Web/CSS/width>

Modèle de boîte – Propriétés "height" et "width"

- La dimension de la balise "span" n'a pas changé :
 - Ces propriétés ne s'appliquent pas sur les balises de type "inline"
- Les propriétés "width" et "height" peuvent être également limitées par les propriétés suivantes :
 - min-height / max-height
 - min-width / max-width

Sources :

- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Le_modele_de_boite
- <https://developer.mozilla.org/fr/docs/Web/CSS/height>
- <https://developer.mozilla.org/fr/docs/Web/CSS/width>
- https://developer.mozilla.org/fr/docs/Web/CSS/Mod%C3%A8le_de_bo%C3%A9te_CSS

Pratiquons ! - Modèle de boîte (Partie 2)

Pré-requis :

- Avoir la ressource ressources/boite

A télécharger ici :

<https://download-directory.github.io/?url=https://github.com/DanYellow/cours/tree/main/integration-web-s1/travaux-pratiques/numero-3/ressources>

Modèle de boîte

Margin – Marge extérieure

Padding – Marge intérieure

Contenu

largeur x hauteur

Modèle de boîte - padding

- Marge intérieure d'une balise
 - Idéal pour mettre de l'espace entre le contenu et le conteneur
- Représentée par la propriété "padding"
- Accepte les valeurs suivantes :
 - Longueur. Exemple : "197px"
 - Pourcentage. Exemple : "25 %"
 - "auto", la navigateur choisit lui-même la dimension

Source :

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

Modèle de boîte - padding

- Ce comportement est lié au fait que le `` est une balise de type "inline".
De fait si le padding est bien visible, l'espace occupé en hauteur par le `` reste son contenu
- Pour une balise de type "inline", il est préférable d'appliquer la propriété "padding" sur les côtés

Source :

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

Modèle de boîte - Padding

padding: 20px;

- La propriété "padding" s'applique à tous les côtés. Toutefois, il est possible de séparer chaque côté grâce aux propriétés :
 - padding-top
 - padding-right
 - padding-bottom
 - padding-left

Source :

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

Modèle de boîte - Padding

padding: 20px;

- On peut associer par côté directement sur la propriété padding, il faut espacer les composantes.
Exemple :

padding: 20px 31% 50px 90%;

| | | |
|---------------------------------------|--|--|
| | padding-right marge intérieure droite | padding-left marge intérieure gauche |
| padding-top marge intérieure haute | | padding-bottom marge intérieure basse |

- Ceci fonctionne également pour la propriété "margin"

Source :

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

Pratiquons ! - Modèle de boîte (Partie 3)

Pré-requis :

- Avoir la ressource ressources/boite

A télécharger ici :

<https://download-directory.github.io/?url=https://github.com/DanYellow/cours/tree/main/integration-web-s1/travaux-pratiques/numero-3/ressources>

Sources :

- <https://code.visualstudio.com/Download>

Et ça a fonctionné ?!

CSS – Spécificité ou poids du CSS

Règle de base : la dernière déclaration CSS s'applique à un élément HTML... mais les règles suivantes surpassent cette règle :

Sources :

- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#Sp%C3%A9cificit%C3%A9
- <https://www.zonecss.fr/cours-css/poids-des-selecteurs-css.html>
- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#!important
- <https://specificity.keegan.st/> - calculateur de spécificité

CSS – Spécificité ou poids du CSS

- Utilisation d'un id (#) comme sélecteur CSS – à éviter si possible
 - Les id ont un autre rôle (voir TP#5)
- Utilisation de l'attribut "style" pour le style

Sources :

- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#Sp%C3%A9cificit%C3%A9
- <https://www.zonecss.fr/cours-css/poids-des-selecteurs-css.html>
- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#!important
- <https://specificity.keegan.st/> - calculateur de spécificité

CSS – Spécificité ou poids du CSS

- Sélecteur CSS plus spécifique
 - Le sélecteur ".infos" est moins spécifique que "div .infos"

Les déclarations css de "div .infos" seront donc appliquées en premier même si placé plus haut dans le code

Sources :

- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#Sp%C3%A9cificit%C3%A9
- <https://www.zonecss.fr/cours-css/poids-des-selecteurs-css.html>
- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#!important
- <https://specificity.keegan.st/> - calculateur de spécificité

CSS – Spécificité ou poids du css - Fusion

```
.infos p {  
  color: blue;  
}
```



```
<article class="infos">  
  <p>Mon texte</p>  
</article>
```

```
.infos {  
  color: red;  
}
```

Le sélecteur ".infos p" est plus précis que ".infos" même si ce dernier est plus bas dans le code, le texte est donc bleu

Sources :

- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#Sp%C3%A9cificit%C3%A9
- <https://www.zonecss.fr/cours-css/poids-des-selecteurs-css.html>
- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#!important
- <https://specificity.keegan.st/> - calculateur de spécificité

CSS – Spécificité ou poids du CSS

- Utilisation de la valeur spéciale "!important"
 - **A utiliser avec extrême parcimonie**
 - S'applique après la valeur d'une propriété CSS :
padding-left: 20px!important;
 - Et lui permet de passer devant toutes les autres règles
 - Se situe en haut de toutes les règles de la spécificité du CSS
 - **A utiliser qu'en dernier cours**

Sources :

- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#Sp%C3%A9cificit%C3%A9
- <https://www.zonecss.fr/cours-css/poids-des-selecteurs-css.html>
- https://developer.mozilla.org/fr/docs/Apprendre/CSS/Building_blocks/Cascade_et_heritage#!important
- <https://specificity.keegan.st/> - calculateur de spécificité

Modèle de boîte

Margin – Marge extérieure

Padding – Marge intérieure

Contenu

largeur x hauteur

Modèle de boîte - Margin

- Marge extérieure d'une balise
 - Idéal pour mettre de l'espace en entre deux balises juxtaposées (verticalement ou horizontalement)
- Représenté par la propriété "margin"

Source :

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

Modèle de boîte - Margin

- Accepte les valeurs suivantes :
 - Longueur (qui peut être négative). Ex: "197px"
 - Pourcentage (qui peut être négatif). Ex: "25%"
 - "auto", la navigateur choisit lui-même la dimension. Permet également de centrer horizontalement, uniquement si la balise a une taille fixe

Source :

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

Modèle de boîte - Margin

- Les marges basses et hautes de balises juxtaposées sont toujours fusionnées (sauf cas spéciaux voir sources), la marge la plus grande des deux est utilisée

Sources :

- <https://developer.mozilla.org/fr/docs/Web/CSS/margin>
- https://developer.mozilla.org/fr/docs/Web/CSS/Mod%C3%A8le_de_bo%C3%A9te_CSS/Fusion_des_marges

Modèle de boîte - Margin

margin: 42px;

- La propriété "margin" s'applique à tous les côtés. Toutefois, il est possible de séparer chaque côté grâce aux propriétés :
 - margin-top
 - margin-right
 - margin-bottom
 - margin-left

Source :

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

Pratiquons ! - Modèle de boîte (Partie 3)

Pré-requis :

- Avoir la ressource ressources/boite

A télécharger ici :

<https://download-directory.github.io/?url=https://github.com/DanYellow/cours/tree/main/integration-web-s1/travaux-pratiques/numero-3/ressources>

Sources :

- <https://code.visualstudio.com/Download>

Modèle de boîte - Margin

- Une balise ayant une largeur (width) et la valeur "auto" pour les propriétés "margin-left" et "margin-right" sera centrée dans son conteneur parent

```
margin-left: auto;  
margin-right: auto;  
width: 440px;
```

Source :

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

Modèle de boîte – Rappel

Margin – Marge extérieure

Padding – Marge intérieure

Contenu

largeur x hauteur

Bordures

- Propriété CSS définie par le terme "border"
- Permet d'afficher une bordure et d'en définir :
 - Couleur
 - Style
 - Épaisseur
- Applique le style sur les quatre côtés de la balise
border: 2px solid blue;

Source :

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

Bordures - Couleur

- Valeur représentant la couleur de la bordure
- Accepte comme valeur, les valeurs d'une couleur habituel (vu précédemment)
- Représente la couleur des quatre côtés de la balise
- Peut-être décorrélié de la propriété "border" avec la propriété "border-color"

Sources :

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







Bordures - Style

- Valeur représentant la style de la bordure
- Accepte comme valeur des valeurs définie (voir sources)
- Représente le style de bordure des quatre côtés de la balise
- Peut-être décorrélé de la propriété "border" avec la propriété "border-style"

Source :

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

Bordures – Style – Exemples de style de bordure

| | | |
|--------|---|---|
| dotted |  | Affiche une série de points ronds. L'espace entre ces points n'est pas défini par la spécification et est donc laissé à l'implémentation. Le rayon des points correspond à la moitié de <code>border-width</code> . |
| dashed |  | Affiche une série de tirets. La taille de ces tirets n'est pas définie par la spécification et est laissée à l'implémentation. |
| solid |  | Affiche une ligne droite continue. |
| double |  | Deux lignes droites sont affichées dont la somme des épaisseurs correspond à la valeur fournie par <code>border-width</code> . |
| groove |  | La bordure crée un effet 3D donnant l'impression qu'elle a été gravée dans le document. On obtient un effet opposé à <code>ridge</code> . |
| ridge |  | La bordure crée un effet 3D donnant l'impression que la bordure ressort du document. L'effet obtenu est opposé à celui obtenu avec <code>groove</code> . |

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

Source :

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

Bordures - Width

- Valeur représentant l'épaisseur de la bordure
- Accepte comme valeur des valeurs définies (voir source) ainsi qu'une longueur
- Représente l'épaisseur des quatre côtés de la balise
- Peut-être décorrélé de la propriété "border" avec la propriété "border-width"

Sources :

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

Bordures

- La propriété "border" peut être décomposée par côté :
 - border-top
 - border-right
 - border-bottom
 - border-left

Sources :

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

Bordures

- Les propriétés "border-width/color/style" peuvent être décomposées par côté :
 - border-top-{propriété}
 - border-right-{propriété}
 - border-bottom-{propriété}
 - border-left-{propriété}

Sources :

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

Bordures

- La propriété "border" a une propriété "similaire" : outline
- L'outline se place au-dessus de la bordure
- La propriété "outline" diffère de "border" en deux points :
 - Ne prend pas de place dans le flux
 - Ne permet pas d'avoir un design différent sur chaque côté

Sources :

- <https://developer.mozilla.org/fr/docs/Web/CSS/border>
- <https://stackoverflow.com/questions/1158515/difference-between-outline-and-border> - en
- <https://developer.mozilla.org/fr/docs/Web/CSS/outline>

Bordures

- Il est possible de mettre une image en bordure avec la propriété "border-image"
 - L'image sera la même sur tous les côtés. La propriété "border-image-left" (par exemple) n'existe pas
- Il est possible de faire des bordures arrondies grâce à la propriété "border-radius"

Sources :

- <https://developer.mozilla.org/fr/docs/Web/CSS/border-image>
- <https://developer.mozilla.org/fr/docs/Web/CSS/border-radius>

Modèle de boîte vraiment complet

Margin – Marge extérieure

Bordure / Outline

Padding – Marge intérieure

Contenu

largeur x hauteur

Modèle de boîte en vrac

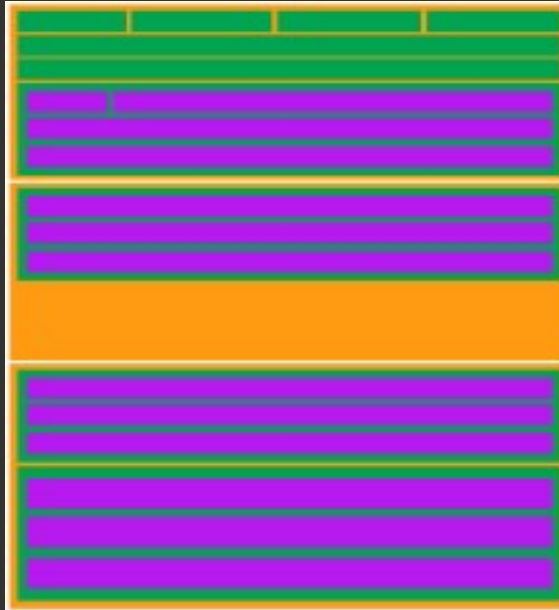
- La propriété "visibility" permet de cacher ou non un élément html tout en conservant sa place dans la page
- La propriété "overflow" permet de gérer le dépassement de contenu

Sources :

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

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 du web, les développeurs utilisaient des tableaux pour réaliser les mise en page
 - Ne jamais, jamais utiliser ceci !
- Ensuite la propriété "float"
 - **A éviter. Crée plein d'effets de bord**

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

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

- Propriété s'applique sur le parent
 - 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>

Pratiquons ! - Découvrons flexbox

Pré-requis :

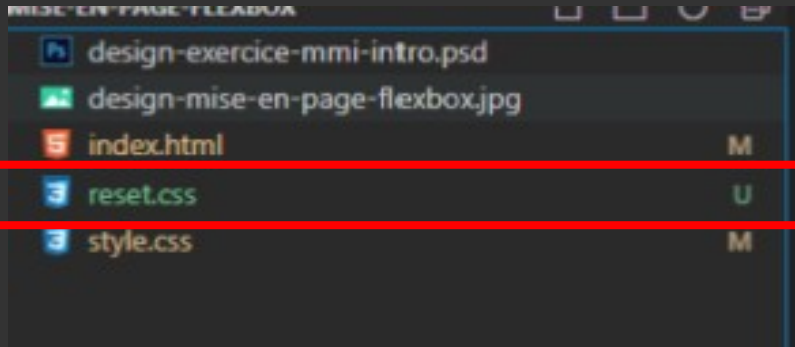
- Avoir la ressource [ressources/site-flexbox/](#)

A télécharger ici :

<https://download-directory.github.io/?url=https://github.com/DanYellow/cours/tree/main/integration-web-s1/travaux-pratiques/numero-3/ressources>

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
- S'oppose à la philosophie du normalize.css
 - Les balises conservent leur style par défaut mais il est uniforme entre les navigateurs

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

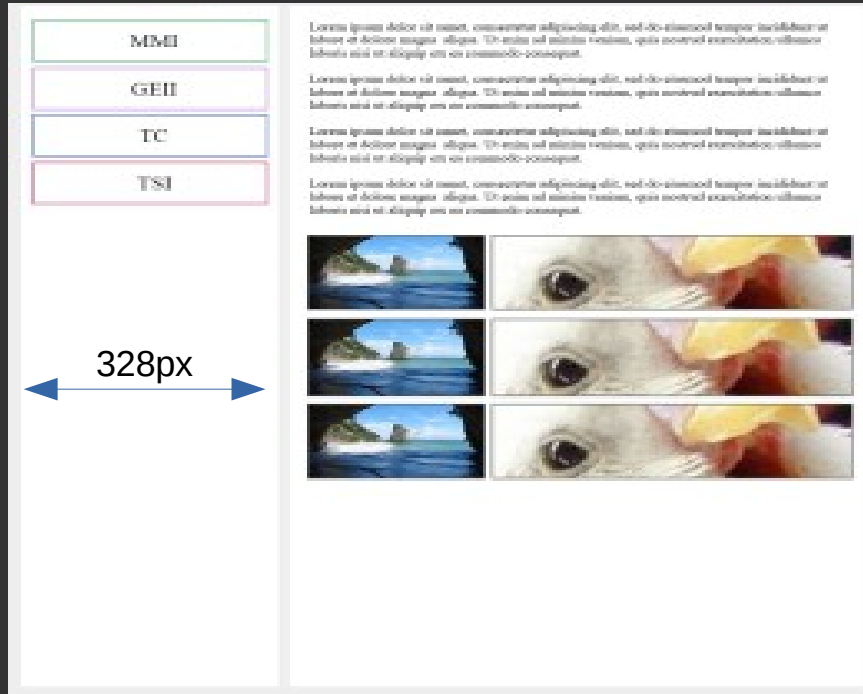
Pratiquons ! - Découvrons flexbox - flex

- "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;  
}
```

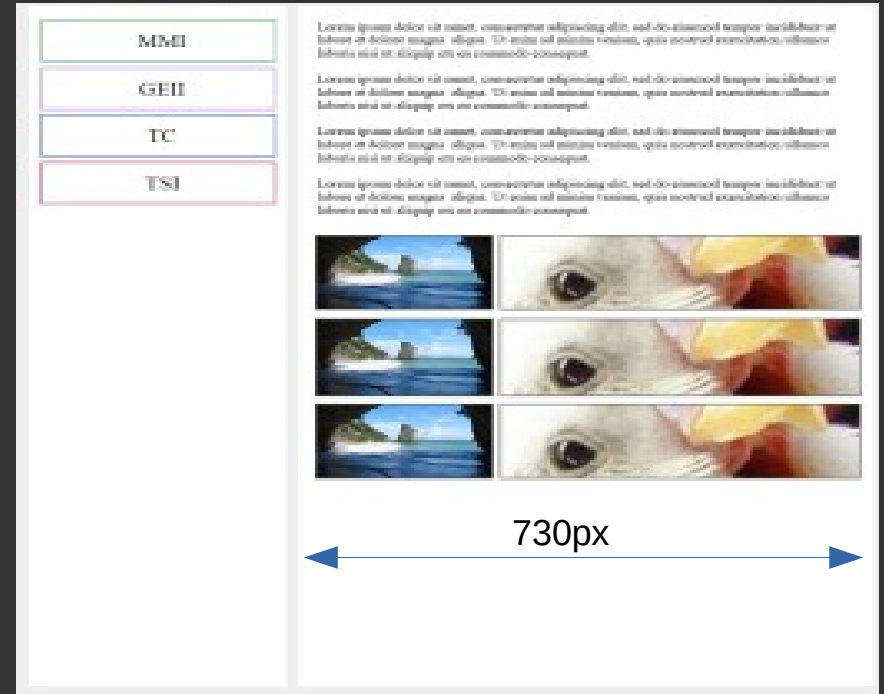
Sources :

- <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

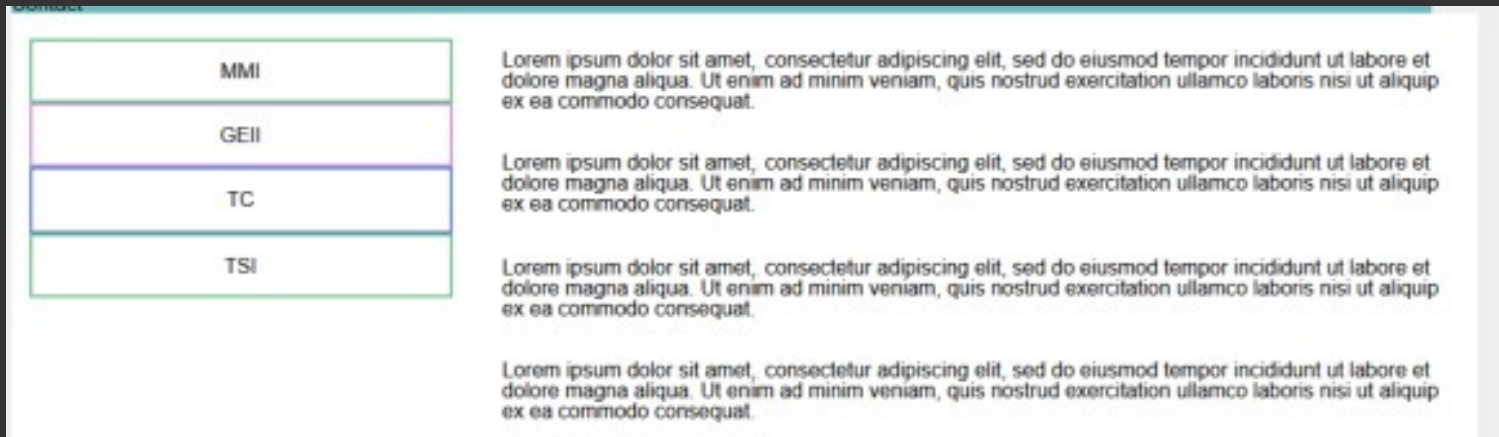
```
.description {  
    flex: 0 0 730px;  
}
```



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

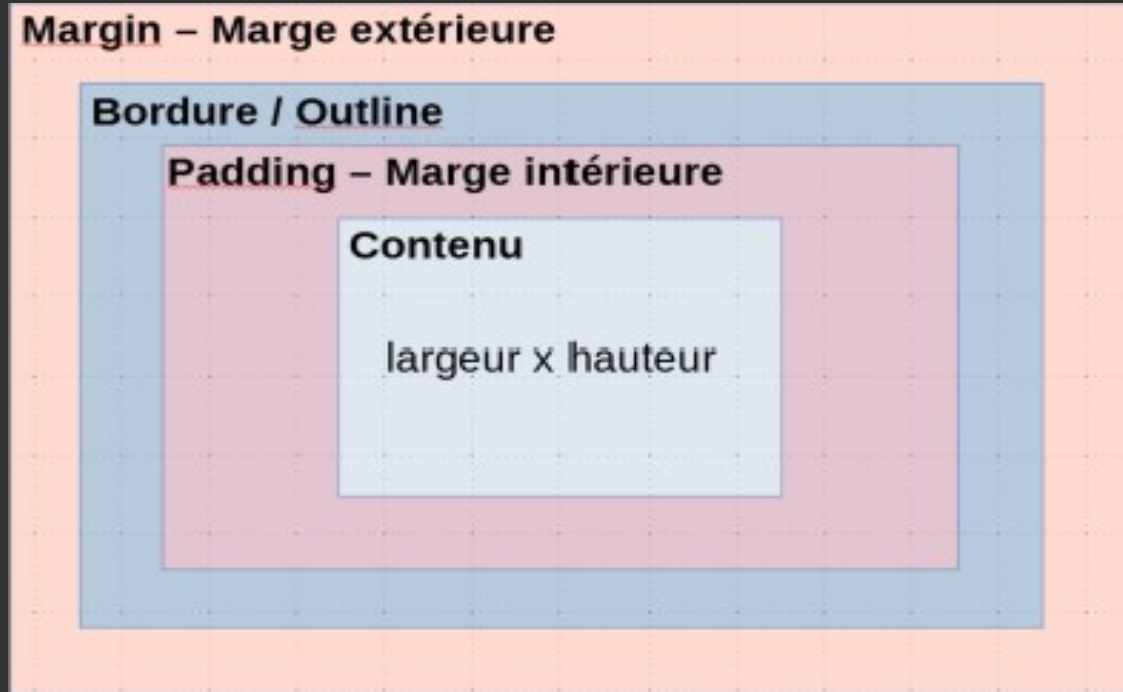
Pratiquons ! - Découvrons flexbox, mise en page



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

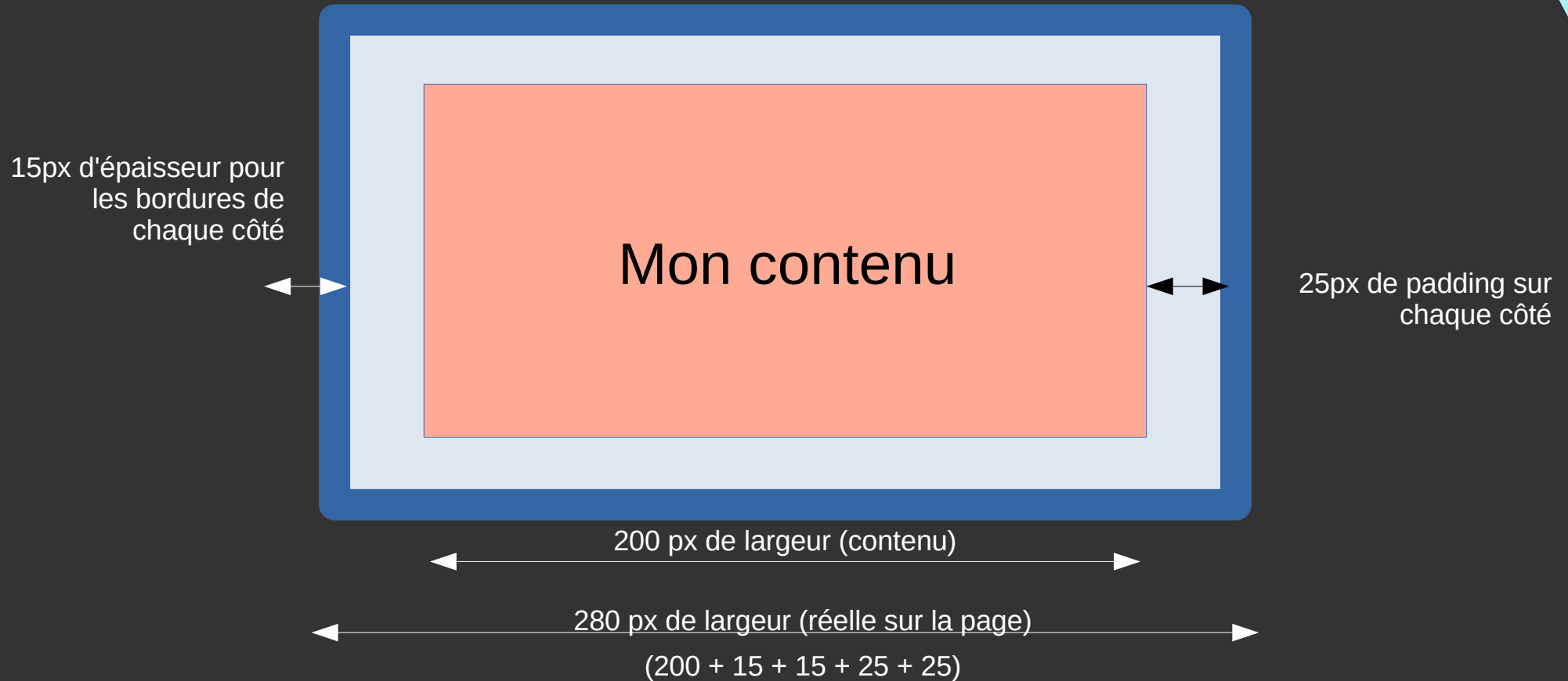


Calcul de dimension de boite

- 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)

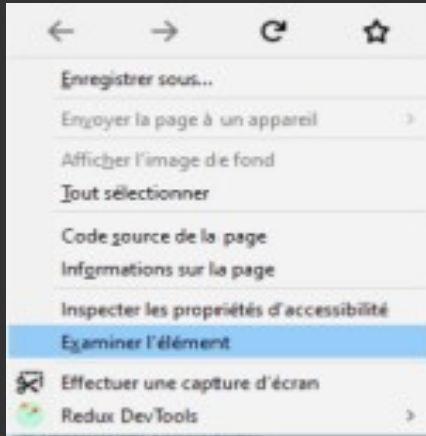
Testons la propriété... sans toucher à VS Code

Console du navigateur

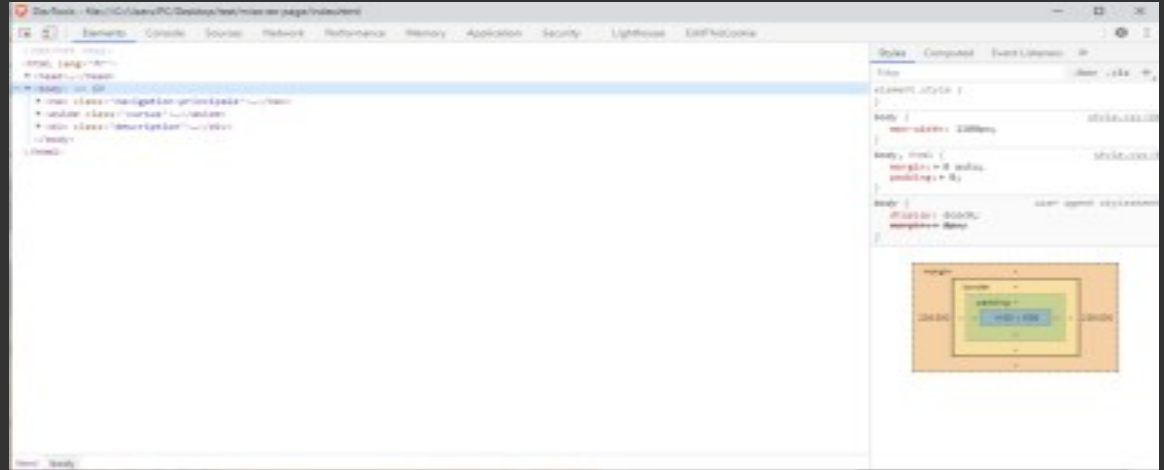
- Outil natif des navigateurs modernes
 - Même sous mobile avec un ordinateur
- Permet d'effectuer rapidement des modifications / tests sur une page web
- Les changements sont locaux
- Indispensable pour débbugger son site
- Permet "d'émuler" des appareils mobiles

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

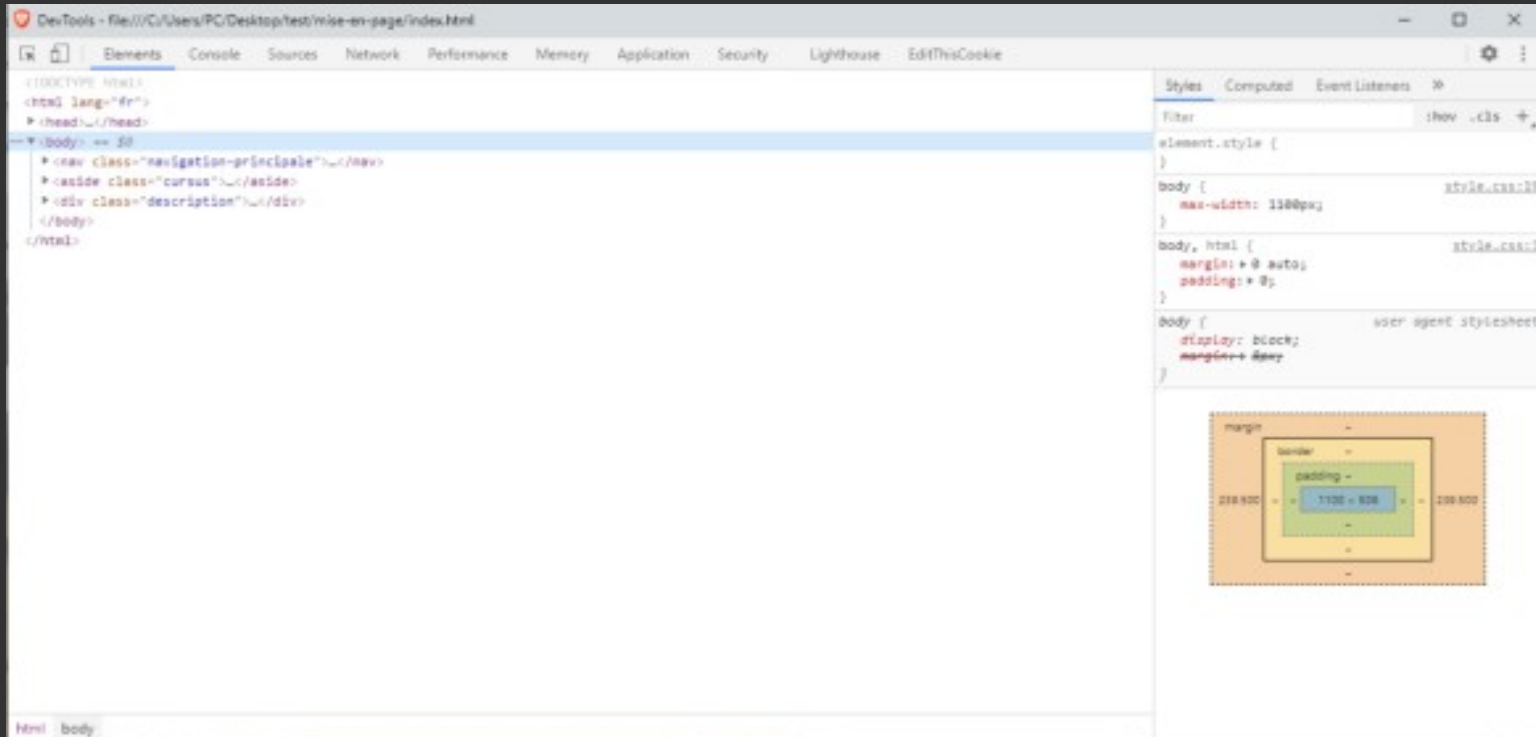


console du navigateur Brave

Sources :

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

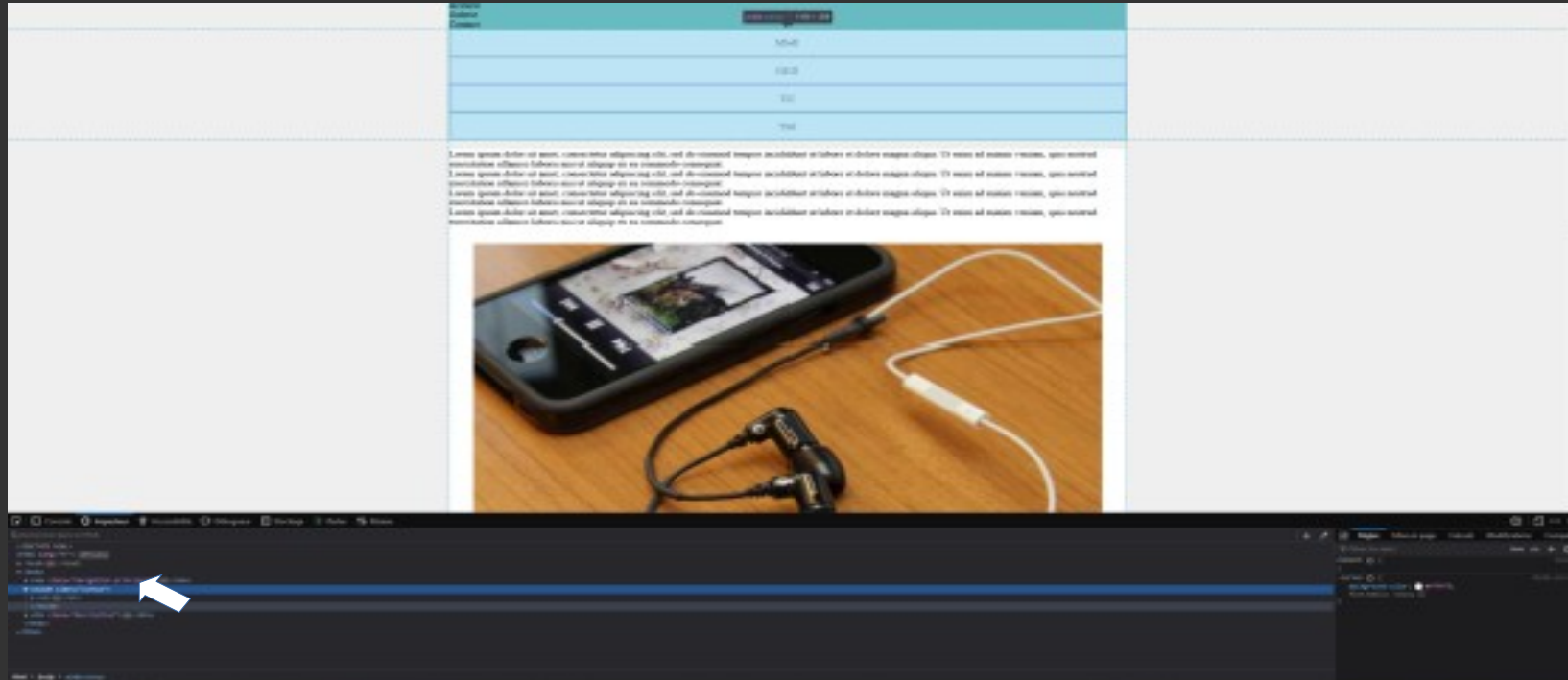
Console du navigateur



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 + [x]

Inherited from h2

```
h2 {
  autooptimize_9388 css:15
  font-size: 30px;
}
h1, h2, autooptimize_9388 css:15
h3, h4,
h5, h6 {
  clear: both;
  font-family: source sans
```

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 :

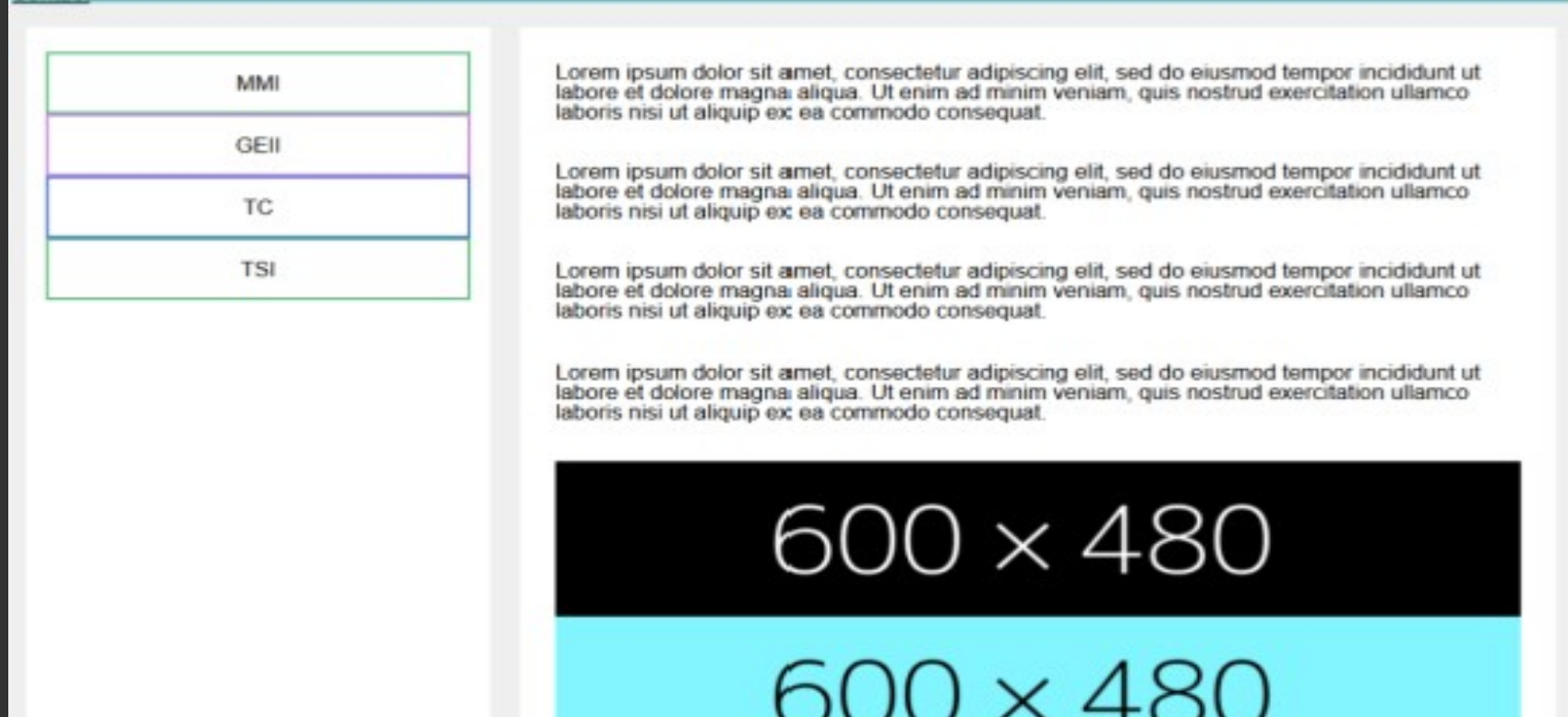
<https://download-directory.github.io/?url=https://github.com/DanYellow/cours/tree/main/integration-web-s1/travaux-pratiques/numero-3/ressources>

- 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



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"

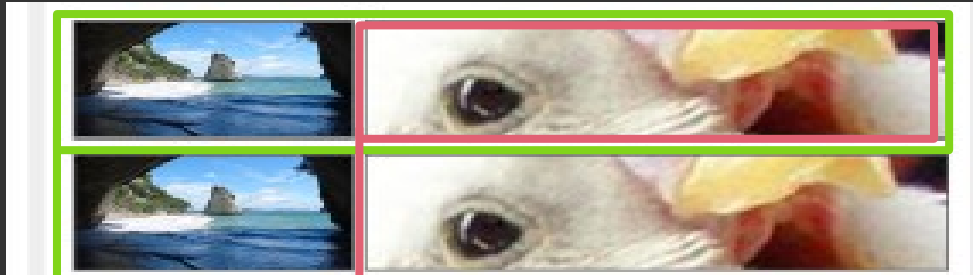
bloc "global" .description

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

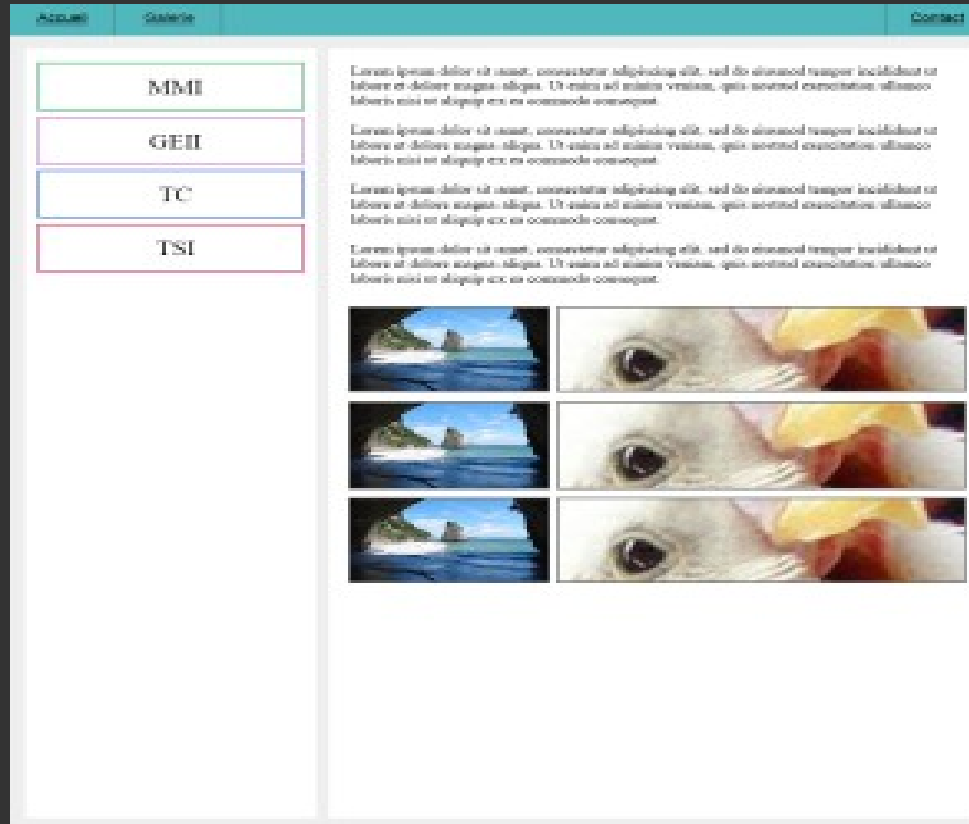


- La propriété flex-wrap indique si les enfants d'un élément flexbox doivent retourner à la ligne
- Possibilité de changer la direction de chaque ligne grâce à la valeur "wrap-reverse"

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 – Flexbox – En vrac

- La propriété "order", sur les enfants d'un élément flexbox, permet d'en définir manuellement l'ordre
- Un enfant d'un élément flexbox peut surcharger l'alignement de la propriété parente "align-items" avec la propriété "align-self"
- Flexbox possède une petite sœur bien plus puissante : "grid". Elle permet de gérer les éléments sur deux axes : horizontal et vertical

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

Pour aller plus loin : "Test - TP#3" sur ENT

Questions ?