***Exercice 14***

*Poursuite du TP sur le site « one page »*



**Petit effet de parallaxe**

Fixer l’image en background de la section « accueil ». Vérifier le rendu en scrollant

**Ajout du contenu dans la section webdesign**

Copier le contenu du ficher « webdesign.html » qui se trouve entre les balises « section » et le coller dans le fichier index.html, dans la section webdesign. Vérifier les différents emplacements des images …

Rappel : figure et figcaption ;

Ce sont des balises html5 que l'on peut utiliser pour la gestion des images et de leur légende (figcaption).

<time> permet de baliser sémantiquement les dates, c’est une balise HTML5.

**Mise en forme de la section « Webdesign »**

**Section** :

* mettre un fond de couleur bleue : #3498db

Vérifier le rendu.

* Ajouter un padding-top de 6em pour « abaisser » le titre h1.

**Le titre de niveau 1 :**

* la taille de la police est de 6em ;
* L’épaisseur (font-weight ) de 700 ;
* la famille de police : Roboto ;
* la couleur d’écriture est #ecf0f1;
* le titre est écrit en majuscule et le texte est centré.

Vérifier le rendu.

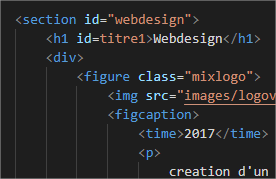
**Alignement des éléments figure :**

Ils doivent être **alignés côte à côte**. Pour cela, vous allez utiliser le « module » **flex**, que vous avez vu lors du précédent TP.

L’application de la propriété « **display : flex** » doit s’effectuer sur le **parent**. Le parent des éléments figure est la **section.**

Essayez d’appliquer **« display : flex » à la section**. **Vérifier le rendu. Que se passe-t-il** ?

Pour résoudre ce problème, il est nécessaire « d’encapsuler » les éléments figure à l’intérieur d’une balise div.



Enlever la propriété « dispaly :flex » à la section, et appliquer à l’élément **div, la propriété « display : flex ».**

Vérifier le rendu.

Cette propriété fonctionne pour aligner les éléments côte à côte, mais le problème est qu’elle les retaille pour les mettre **tous alignés sur la même ligne, sans retour.**

Si on souhaite le retour à la ligne, il faut ajouter la propriété « **flex-wrap: wrap;** »

Vérifier le rendu.

Pour centrer le contenu, essayez d’appliquer, toujours pour la « div », la propriété « text-align : center ».

Vérifier le rendu. (Ce n’est pas centré…)

**Si cela ne fonctionne pas !** Pour centrer les éléments enfants d’un « display : flex », il faut utiliser la propriété « **justify-content : center** ».

**Les éléments figure :**

Ils ont une largeur de 240px, une hauteur de 150px, avec une bordure de 3px, en trait continu et de couleur #ecf0f1,

La taille de l’élément figure correspond à la taille de l’image.

**Figcaption :**

Font-family : lucida sans

Font-size :8px

Couleur : blanche

On souhaite cacher le texte, pour cela il est nécessaire de cacher ce qui dépasse

(overflow :hidden).

**Au survol de la figure**, « faire remonter » la **figcaption** de 70px à l’aide d’une transformation (translateY).. La couleur d’écriture est blanche, le texte est centré. **Appliquer une transition de 0.5s sur la transformation.**

Ajouter une propriété permettant **d’agrandir l’image** lorsqu’elle est survolée. Appliquer pour cette transformation, une **transition de 0.5s**.

Height : 240 px

Width : 220px

Alignement du texte.

**Header fixe**

* Mettre le **header en position fixe.**
* Vérifier le rendu.
* L’élément ne fait plus partie du **flux**,

il faut donc repréciser certains **éléments de taille**.

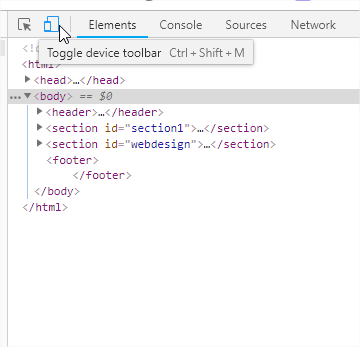
* Ajouter, pour le header, une **largeur à 100%**, modifier les valeurs du **padding à 5px 10px** (au lieu de 5px 30px)
* Scroller la page : pour éviter que les éléments transformés ne passent au-dessus du menu, ajouter un z-index :100 pour le header.
* Ajuster la section1

**Affichage en responsive :**

Lorsqu’on réduit la taille de l’écran, on constate que les éléments se chevauchent et que le tout n’est pas très esthétique ni fonctionnel.

A l’aide de **Chrome**, visualiser le rendu avec différentes résolution.

Pour cela, afficher « l’inspecteur » (touche F12). Puis cliquer sur la touche « **Toggle device toolbar** » :

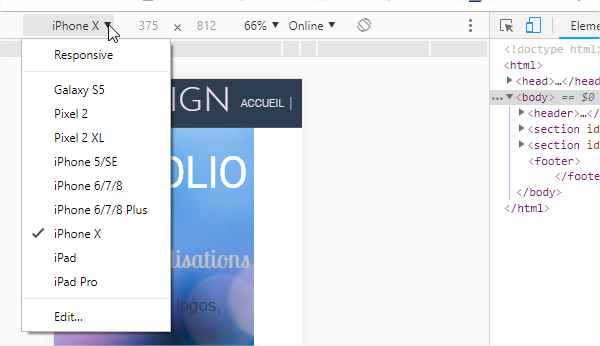
Vous avec alors accès à des poignées qui vous permettent de choisir les tailles d’affichage en pixel.

On peut ainsi constater qu’il y a des problèmes d’affichage du titre.

En cliquant sur

MMI - HTML/CSS

*Elisabeth Pécatte année 2016-2017* ***3 |*** *P a g e*

En cliquant sur « Responsive », il est également possible de choisir l’appareil et visualiser le rendu suivant le type de  portable

(*Remarque : pour revenir à un affichage « normal », il faut recliquer sur l’outil)* 

- Dans le fichier HTML :

o **Ajouter la balise meta name « viewport »**.

Vérifier de nouveau l’affichage avec l’outil « Toggle device ».

Vous remarquez que l’outil ne dé-zoome plus et que l’affichage dans l’outil de Chrome correspond à l’affichage lors de la diminution de la fenêtre du navigateur.



**Ajouter une MediaQuery pour les écrans qui font moins de 950px.**

Le menu ne sera plus affiché en ligne mais il sera affiché en bloc.

Pour cela :

* **Pour le menu** : mettre un fond de couleur bleu sombre #2c3e50.
* **Pour les items du menu :** les afficher en bloc, avec une hauteur (height) de 2em, et un text-align à gauche.
* **Pour les liens du menu :** ils n’ont pas de bordure à droite.

**Menu en responsive**

**Création d’un bouton « ouvrir/fermer ».**

Dans le fichier HTML, créer une ancre en ajoutant un id au body :



Ajouter ensuite deux liens :

* Un qui pointe vers la page en cours (href="#") avec la classe nav-close,
* et l'autre qui pointe vers l'ancre créée précédemment

(href="#contenu")avec la classe nav-open .



***Au niveau du CSS***

* Placement des liens de classe nav-open et nav-close en position **fixed**, en haut et à droite (dans la partie des média query)



* Ouvrir et fermer sont positionnés au même endroit et se chevauchent donc.
* On ne va rendre qu’un seul de ces deux liens visible. Par défaut, c'est le lien "ouvrir" qui est visible et « fermer » est invisible. Ajouter donc les règles suivantes.



Dans la partie du fichier css, hors mediaquery, rendre les deux boutons invisibles.

* Pour afficher le bouton fermer (et rendre le bouton « ouvrir » invisible),

il faut utiliser la pseudo classe :target qui permet de cibler une ancre active.



Tester en cliquant successivement sur « ouvrir » et fermer » et en vérifiant l’URL.

L’étape suivante consiste à masquer le menu.

Pour cela, il va être positionné à « l’extérieur » de l’écran, à 300px sur la droite. On va également le « descendre » de 5em pour qu’il se positionne sous le header.

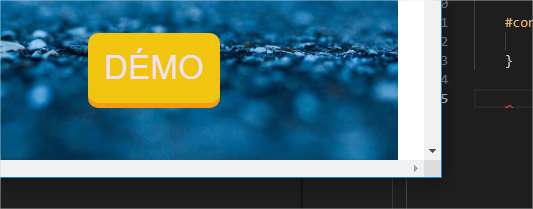


Il faut ensuite afficher le menu, et pour cela, il faut de nouveau utiliser le pseudo classe target.



Tester

On peut remarquer que le navigateur ajoute un scroll. En effet, par défaut, le navigateur affiche tout le contenu.



Si c’est bien votre cas :

Pour supprimer ce scroll, ajouter dans le CSS, tout en haut du fichier, pour le sélecteur « **body** », la propriété « **overflow-x :hidden** »

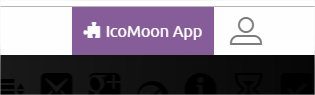
**Modification des boutons « ouvrir » et « fermer » en bouton hamburger.**

Il existe plusieurs méthodes pour afficher un menu « hamburger ».

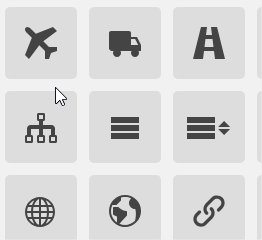
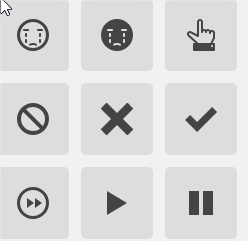
Nous allons utiliser une police de caractères.

Pour cela, cliquer sur le site [https://icomoon.io/](https://icomoon.io/%20)

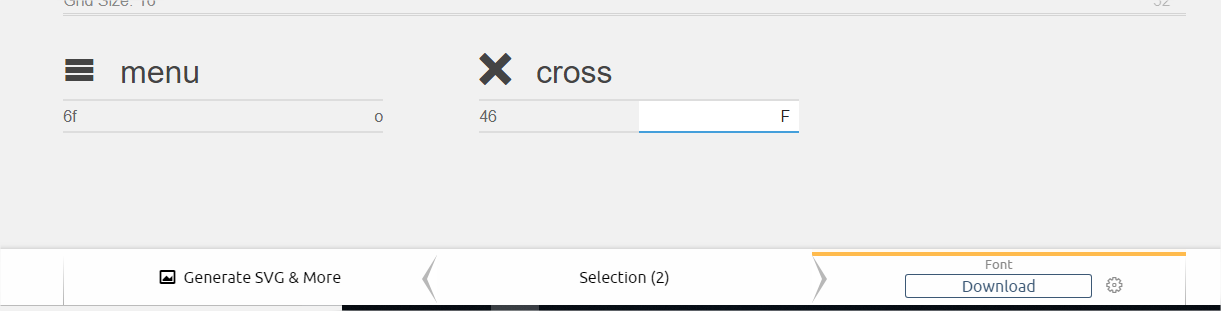
Puis sur le bouton IcoMoonApp en haut à droite.



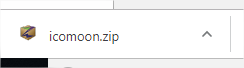
Chercher et cliquer sur l’icône « hamburger » et sur l’image X :

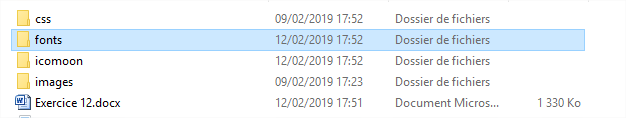
Lorsque les deux icônes sont sélectionnés, générer une police de caractères en cliquant en bas à droite sur « Generate font »



Pour le hamburger, choisir la lettre « O » (pour ouvrir) et pour la croix « F » pour fermer.

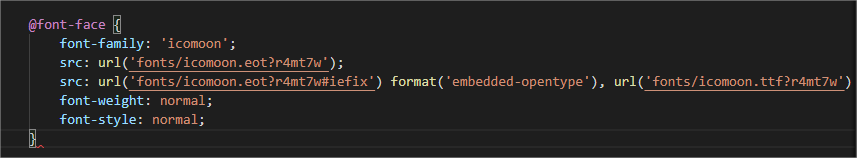
 Cliquer ensuite sur « Download » et enregistrer le fichier.

Dézipper le fichier téléchargé, et placer le dossier « fonts » dans le dossier theme ( au même niveau que « css » et « images »)



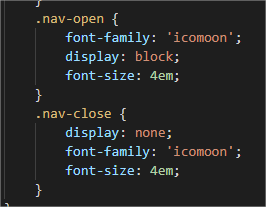
A l’aide de @font-face, déclarer la police de caractère

(Vous pouvez copier la déclaration présente dans le fichier style.css présent dans le zip, mais attention, il faut penser à modifier le chemin vers les fontes).



Puis utilisez cette famille de police pour les classes nav-open et nav-close.

Mettre la taille de la police à 4em.



Dans le fichier HTML, écrire un « O » à la place de « ouvrir » et un « F» à la place de « fermer ».



Observer le rendu.

