

Exercice 13

Le but de cette séance est la création d'une page Web sur laquelle vous serez amenés à travailler plus tard pour la compléter.

A la fin de cette séance, vous devriez obtenir cette page :



Récupérer sur Moodle le dossier exercice11seguier qui contient un fichier index.html, un dossier « images » qui contient les images nécessaires au thème.
Créer un dossier « css ». Créer un nouveau fichier « style.css » dans le dossier « css ». Relier le css au fichier HTML.

Le body :

- ✓ La police utilisée est Arial.
- ✓ La taille de la police est de 0.8em. *
- ✓ La hauteur de ligne est de 1em
- ✓ Les marges internes et externes sont à 0.

*em est une unité de mesure qui correspond à la taille de police choisie par l'internaute dans son navigateur. Si rien n'est changé, la valeur par défaut est 16px. Cette valeur est réglable dans les paramètres du navigateur.

//////////////////// Mise en forme du header //////////////////////

Le header :

- ✓ La couleur de fond est #2c3e50
- ✓ Les marges internes sont de 5px en haut et en bas, et de 30px à droite et à gauche.
- ✓ La hauteur est de 5em.

Le titre h1 dans le header :

La police de caractère est une police GoogleFont 'Josefin Sans'. Il y a deux épaisseurs de caractères qui seront utilisées pour le titre, 300 et 700, il faut donc préciser les épaisseurs souhaitées dans googlefont tel que :

Une fois la police sélectionnée, cliquer sur l'onglet « CUSTOMIZE »

Sélectionner 300 et 700

Recliquer ensuite sur l'onglet « EMBED » et copier le lien dans le fichier HTML

- La police est Josefin Sans
- L'épaisseur de la police est de 300
- La taille est de 4em
- La couleur d'écriture est rgb (241,225,245)
- Les marges externes sont de 0

Travail sur le W du logo :

Dans le fichier HTML, encadrer le W à l'aide d'une balise span et spécifier une classe.

```
<h1><span id= "logo">W< /span>ebdesign</h1>
```

Dans le fichier css, ce span a :

- Une couleur d'écriture de #9b59b6
- Une épaisseur de caractère de 700
- Ajouter une taille de police de 4em, que se passe-t-il ?
 - On peut constater que les 4em affectés au span, enfant de h1, n'ont pas la même valeur que les 4em affectés au h1.
 - Comment l'expliquez-vous ?
- Essayer une hauteur et une largeur de 1.2em.
 - Regarder le rendu sur le navigateur. Pour le décaler vers le bas, ajouter une hauteur de ligne de 1.4em.

//////////////////// Mise en forme du menu //////////////////////

Dans le fichier HTML, créer les ancres :

- Le lien « Accueil » vers la « section1 »
- Le lien « Webdesign » vers la « section2 »
- Le lien « Ressources » vers la « section3 »
- Le lien « Contact » vers la « section4 »

Aligner les items de liste côte à côte ... (display : flex ;)

Sans ajouter d'id ou de class dans le fichier HTML, ajouter dans le fichier css les styles suivants :

Pour la liste :

- La famille de police est la font google « roboto »,
- Les marges internes et externes sont égales à 0,
- Le texte est centré,

Pour les items de liste :

- Retirer les puces,

Pour les liens des items de liste :

- Enlever les soulignés,
- Ecrire en rgb (241,225,245) et en Majuscule,
- Police de 1.3em ,
- Un padding à 0 en haut et en bas et de 10px à droite et à gauche,

- Mettre une bordure à droite de 1px, en trait continu, de couleur rgba (241,225,245) transparence de 0,2
- Le lien du dernier item n'a pas de bordure.
- La hauteur de ligne est de 4em (cela permet d'agrandir la bordure, et la zone cliquable)

//////////////////// Lien vers les réseaux sociaux //////////////////////

Dans le fichier CSS:

A l'aide de « display : flex », afficher les items de liste côte à côte.

Pourquoi c'est déjà en place ? ...

Pour la liste :

Ajouter un « padding-left :0 » pour supprimer le padding mis par défaut par le navigateur.

Pour les items de liste :

Pour éviter que les images soient collées l'une à l'autre, ajouter une marge externe à droite de 15px.

//////////////////// Alignement des éléments du header //////////////////////

Les différents éléments du header doivent être affichés les uns à côté des autres.

Utiliser un display :

Autres mises en forme :

- Pour centrer les enfants d'un container auquel un display : flex est appliquée, il existe 2 propriétés :
 - Pour l'alignement sur l'axe principal (ici vertical) : align-items, qui peut prendre, entre autres, les valeurs

```
/* Mots-clés de base */  
align-items: normal;  
align-items: stretch;
```

```
/* Alignement géométrique */
/* align-items ne gère pas les valeurs left et right */
align-items: center; /* Les éléments sont regroupés au centre */
align-items: start; /* Les éléments sont regroupés au début */
align-items: end; /* Les éléments sont regroupés à la fin */
align-items: flex-start; /* Les éléments flexibles sont regroupés au début */
align-items: flex-end; /* Les éléments flexibles sont regroupés à la fin */
align-items: self-start;
align-items: self-end;
```

- Pour l'alignement sur l'axe secondaire (ici horizontal) : C'est la propriété justify-content qu'il faut utiliser :

```
/* Alignement « géométrique » */
justify-content: center; /* Éléments groupés au centre */
justify-content: start; /* Éléments groupés au début */
justify-content: end; /* Éléments groupés à la fin */
justify-content: flex-start; /* Éléments flexibles groupés au début */
justify-content: flex-end; /* Éléments flexibles groupés à la fin */
justify-content: left; /* Éléments groupés à gauche */
justify-content: right; /* Éléments groupés à droite */
```

```
/* Alignement distribué */
justify-content: space-between; /* Les éléments sont répartis équitablement
Le bord du premier est aligné sur le
début du conteneur et la fin du dernier
est alignée sur la fin du conteneur */
justify-content: space-around; /* Les éléments sont répartis équitablement
À chaque extrémité, entre le bord du
conteneur et le premier/dernier élément
on a la moitié de l'espace appliqué entre
chaque élément */
justify-content: space-evenly; /* Les éléments sont répartis équitablement
Tous les éléments sont séparés par le même
espace */
justify-content: stretch; /* Les éléments sont répartis équitablement et
les éléments dimensionnés avec 'auto' sont
étirés afin de remplir le conteneur */
```

Appliquer pour le header, align-items : center ;

Le h1 et le ul doivent être maintenant centré verticalement.

Pour le ul du menu, appliquer justify-content :center ; pour centrer le contenu

Pour la liste des réseaux sociaux, appliquer justify-content :flex-end pour aligner à droite

Dès que l'on utilise la propriété `display : flex` sur un parent, les trois éléments enfant (`h1`, `ul` et `ul`) se mettent côte à côte avec la même largeur.

Diminuer la fenêtre du navigateur et observer ce qui se passe. Il est possible de définir la largeur des éléments enfants ; soit en utilisant `width` (par exemple 20%) soit en utilisant la propriété `flex` et un nombre « ratio ».

Pour `h1` le `ul` des réseaux sociaux, écrire la propriété `flex :1`.

Pour le `ul` du menu, mettre la propriété `flex` à 2.

Observer le rendu.

//////////////////// Mise en forme de la première section //////////////////////

- Appliquer l'image de fond « **background.jpg** ». Attention au chemin ...
- Il faut « agrandir » la section, en mettant une largeur à 100%.
- Appliquer une hauteur de 100% aussi. **Observer le rendu.** (le 100% en hauteur ne fonctionne pas)
- Mettre la hauteur de toutes les sections à **100vh** et observer le rendu.
Positionner l'image de fond au centre et en bas. C'est déjà mieux.
- La taille de l'image de fond doit s'adapter et couvrir toute la section sans être déformée (**background-size**).

On remarque que le texte est collé tout en haut, or on souhaite qu'il soit centré horizontalement et verticalement. Ajouter donc, à la section, les propriétés **text-align : center** et **vertical-align : middle**.

L'alignement vertical ne fonctionne pas ; en effet, la propriété « `vertical-align` » avec la valeur `middle` ne fonctionne pas avec toutes les balises et propriétés.

Pour centrer verticalement le bloc :


- Soit on utilise les valeurs "tabulaires" du `display` (`table`, `table-row`, `table-cell`)
- Soit on utilise la propriété `flex`.
- Appliquer la propriété « `display :flex` » à la section.
 - Tous les éléments enfants sont alignés côte à côte. En effet, par défaut, les éléments sont alignés en `inline-flex`.
- Pour aligner les éléments les uns en dessous des autres, il faut appliquer la propriété : « `flex-direction :column` »

- ////////////////////

- Par défaut, le navigateur applique une marge externe en haut et en bas du h1. Pour

- ////////////////////

- Les traits obliques seront réalisés plus loin dans le tp

- 






- ✓ Sur le survol du lien, inverser les couleurs de fond et de l'ombre.
- ✓ Ajouter une transition de 0,5s

////////////////////////// Les effets sur les textes ////////////////////////////

Rotation dans le H1 de la section (nom et 2019)

Dans le fichier HTML, le mot « mon » est encapsulé dans la balise span avec une class « tourneg » et le mot « 2019 » dans un span avec la class « tourned »

Pour les spans des deux classes, il faut appliquer :

- Une hauteur de ligne de 0.5em et une hauteur de 0.7em
- Une bordure en bas de 0.1em d'épaisseur, un trait continu et de couleur blanche
- L'épaisseur de la police est de 100. Attention pour la police Roboto, il faut faire la même chose manipulation que celle effectuée pour Joselin+Sans concernant les épaisseurs chargées, on peut également modifier directement le fichier HTML en ajoutant les deux tailles comme ceci :

```
<link href= »https://fontsgoogleapiscom/css ?family=Roboto:300,100" rel="stylesheet">
```

```
<link href= »https://fontsgoogleapiscom/css ?family=Joselin+sans:300,700" rel="stylesheet">
```

Pour le “tourneg”

- La taille de la police est de 0.5em
- Appliquer une transformation de rotation de -90 degrés. Observer le rendu.
- Ajouter une translation de 0.5em sur l'axe des X et 0.5em sur l'axe des Y pour rapprocher « mon » de portfolio et pour remonter un peu

Pour le “tourned”

- La taille de la police est de 0.4em (les chiffres étant écrits plus gros par défaut, il faut diminuer la taille par rapport au text « mon » pour un meilleur rendu.
- Appliquer une transformation de rotation de 90 degrés. Observer le rendu.
- Ajouter une translation de -0.6em sur l'axe des X et 0.5em sur l'axe des Y pour rapprocher « mon » de portfolio et pour remonter un peu.

Il n'y a pas de rotation ?

A vous de réfléchir ...

//////////////////// Les traits « obliques » avant et après le paragraphe////////////////////

Pour ajouter du contenu avant et après la paragraphe, il faut utiliser les pseudo classes before et after.

- La couleur de fond est #2c3e50 ;
- La largeur et la hauteur sont de 1em.
- Border 2px solid yellow

Test

Utilisation de la propriété transform :

Appliquer la fonction skewX(-30deg).

Vérifier le rendu...

Essayer de déterminer les raisons du dysfonctionnement.

Penser à utiliser le :hover ...