
TP5 : CSS -Flex

Maxime LORENTZ

Nous allons voir aujourd'hui

1. Concevoir une grille responsive
2. Les médias queries

Exercice 1 : Création d'une grille simple 2 colonnes

Objectif : Débuter une grille simple sur 2 colonnes

Balises utilisées dans l'exercice :

`<style><body><head><div>`

Attributs html utilisés :

`class`

Propriétés css utilisés :

`flex, margin, padding, display, box-sizing, background-color, :first-of-type, :last-of-type, @media`

<https://www.w3.org/TR/css-flexbox-1/>

Html

- Réaliser un document HTML
- Créer une div portant la classe « grille »
- A l'intérieur du div.grille ajouter 2 div portant la classe « col »

Css

- body avec les attribut margin et padding défini à 0
- la classe .grille est un élément flex avec une marge haute et basse de 2 rem
- tous les éléments continus directement dans la grille ont un padding de 2 rem
- toutes les classes .col doivent avoir la même taille **flex:1**, il faudra créer un gouttière entre chaque colonne de 1rem à gauche et droite, une couleur de fond = #D1D1D1
- le premier élément ayant la classe .col a une marge à gauche égale à 0
- le dernier élément ayant la classe .col a une marge à droite égale à 0

Résultat attendu :

Grille 2 colonnes



Exercice 2 : Rendre responsive la grille de 2 colonnes

Objectif : Rendre accessible une mise en page sur mobile

Balises utilisées dans l'exercice :

```
<style><body><head><div>
```

Attributs html utilisés :

```
class
```

Propriétés css utilisés :

```
flex, margin, padding, display, box-sizing, background-color, :first-of-type, :last-of-type, @media
```

On va repartir du premier exercice. On veut que la grille passe en colonne à partir de 640px de large.

Pour cela nous allons utiliser les mediaqueries.

<https://www.w3.org/TR/css3-mediaqueries/>

Les critères utilisés dans les Media Queries sont variés. La plupart de ces critères peuvent être préfixés par "min-" ou "max-" pour définir respectivement une valeur "minimum ou égale" ou "maximum ou égale". Ces critères sont les suivants:

- **Width** : largeur de la zone d'affichage dans le navigateur
- **Height** : hauteur de la zone d'affichage dans le navigateur
- **Device-width** : largeur de l'écran de l'appareil
- **Device-height** : hauteur de l'écran de l'appareil
- **Orientation** : orientation de l'appareil (valeur possible : "portrait" ou "landscape").
- **Aspect-ratio** : ratio de la zone d'affichage du navigateur (rapport "largeur" contre "hauteur"). La valeur est constituée de 2 entiers séparés par un slash (exemple : "16/9")
- **Device-aspect-ratio** : ratio de l'écran de l'appareil (rapport "largeur" contre "hauteur"). La valeur est constituée de 2 entiers séparés par un slash (exemple : "16/9")
- **Color** : nombre de bits par composante de couleur
- **Color-index** : nombre d'entré dans la table de couleurs [indexées](#)
- **Monochrome** : nombre de bits par pixels sur un périphérique monochrome (c'est-à-dire le niveau de gris). Si le périphérique utilise des couleurs, la valeur est zéro
- **Resolution** : densité de pixels du périphérique. La valeur peut être définie en points par pouce (dpi) ou en points par centimètre (dpcm)
- **Scan** : processus de balayage sur une télévision. Valeur possible "progressive" ou "interlace"
- **Grid** : définit si l'appareil est un périphérique en grille ou bitmap

exemple :

```
@media screen and (max-width: 500px) {  
— mettre ici les nouveaux comportement css  
}
```

Dans l'exemple ci-dessus, nous ciblons les écrans dont la largeur d'écran maximum est de 500px

Css

- Ciblez les écrans ayant une largeur maximum de 640px
- Passez la direction du conteneur flex en column
- Supprimer les marges gauche et droite et ajouter une marge basse de 1rem

Grille 2 colonnes

.col

.col

Exercice 3 : Réaliser une grille sur 3 colonnes

Objectif : Manipuler les éléments flex et savoir les dimensionner

Grille 3 colonnes



Propriétés css utilisés :

flex-shrink, flex-grow, flex-basis, order

Consigne :

- Créer une grille homogène avec des gouttières égales.
- En mobile (media query), je souhaite que tous les éléments utilisant un double espacement soit positionné en première position

Exercice 4 : Sans explication je souhaite voir cela

Grille 4 colonnes

.col	.col	.col	.col
.col-2-4		.col	.col
.col	.col-2-4		.col
.col	.col	.col	

Exercice 5 : Si vous êtes arrivé là vous pourrez faire ça :

Grille 2/5 colonnes

.col-2-5		.col	.col	.col
.col	.col-2-5		.col	.col
.col	.col	.col-2-5		.col
.col	.col	.col	.col-2-5	
.col	.col-2-5		.col-2-5	
.col-2-5		.col	.col-2-5	
.col-2-5		.col-2-5		.col

Grille 3/5 colonnes

.col-3-5			.col	.col
.col	.col-3-5			.col
.col	.col	.col-3-5		

Grille 4/5 colonnes

.col-4-5				.col
.col	.col-4-5			

et ça