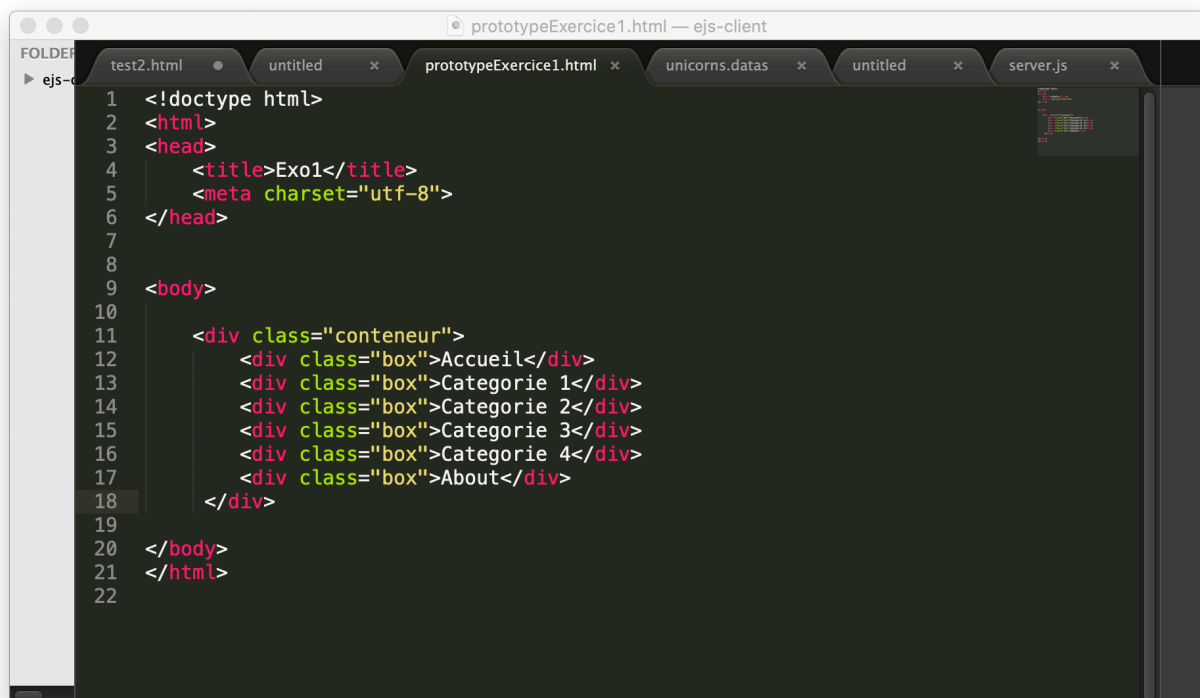


Exercice 1 - HTML / CSS

Exercice à effectuer sur ordinateur avec connexion internet et tous supports autorisés. Cette partie concerne uniquement le CSS de base sans se préoccuper du responsive, sans utiliser grid , flex ou bootstrap. Ici on se préoccupe plutôt des flottants et des positions. Tapez dans votre IDE le code HTML/CSS suivant



```
1 <!doctype html>
2 <html>
3 <head>
4   <title>Exo1</title>
5   <meta charset="utf-8">
6 </head>
7
8
9 <body>
10
11   <div class="conteneur">
12     <div class="box">Accueil</div>
13     <div class="box">Categorie 1</div>
14     <div class="box">Categorie 2</div>
15     <div class="box">Categorie 3</div>
16     <div class="box">Categorie 4</div>
17     <div class="box">About</div>
18   </div>
19
20 </body>
21 </html>
22
```

Afficher le résultat sans style dans un navigateur.

Effectuer un link avec le fichier style.css suivant se trouvant dans un répertoire *asset* à la racine:

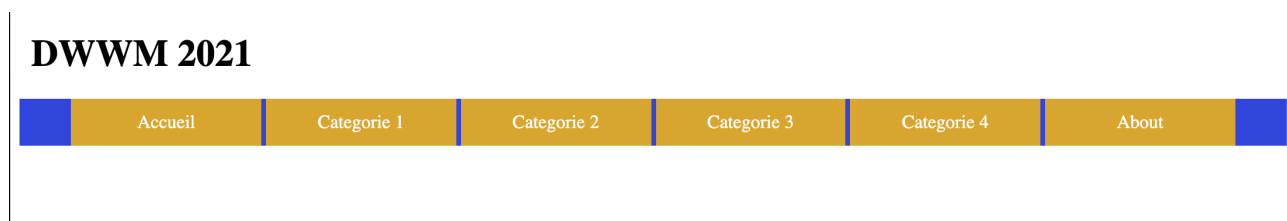
```
body {  
  
}  
div {  
    text-align: center;  
    color: #fff;  
}  
.box {  
  
    background: #dba634;  
    height: 40px;  
    width: 100px;  
}
```

1) Rajouter un titre h1 avant le div.conteneur avec le titre "DWWM 2021"

2) Appliquer une marge de 5px en haut et en bas et 10px à gauche et à droite sur ce titre, un bord avec un trait noir et une distance entre le contenu et le bord de 4px.

3) Donner une largeur de 900px au body et centrer le body

4) Que rajouter dans la classe box pour avoir ce résultat ?



5) En css , au survol des box , changer le fond de ces box en bleu

6) Rajouter un bloc <aside> d'une hauteur de 400px et d'une largeur de 15% à gauche qui de colle au menu avec une couleur de fond.

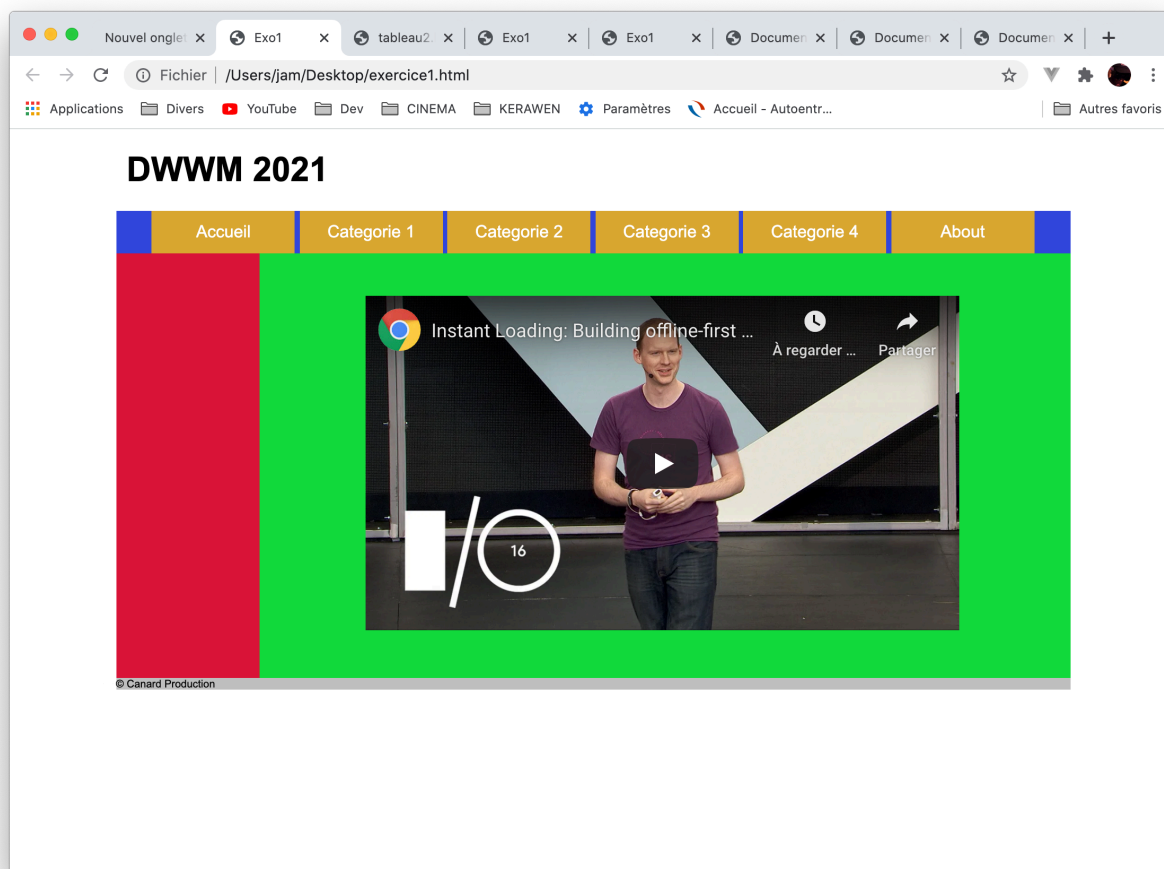
7) Puis un bloc <article> d'une largeur de 85% de la même hauteur qui se colle à droite et au menu avec une couleur de fond.

8) Rajouter un footer avec ce texte (taille 10px) avec un background de couleur silver.
© Openlab Production

9) la balise <iframe> permet d'intégrer une autre page web dans notre page . Intégrer l'iframe suivant dans <article> à 40px du haut et de 100px de la gauche dans l'article :

```
<iframe width="560" height="315" src="https://www.youtube.com/
embed/cmGr0RszHc8" frameborder="0" allow="accelerometer; autoplay;
clipboard-write; encrypted-media; gyroscope; picture-in-picture"
allowfullscreen></iframe>
```

Résultat attendu (hors couleurs de votre choix):



RESPONSIVE

Bootstrap possède des classes prédéfinies qui permettent créer des grilles qui sont créées de manière proportionnelle se basant sur le principe que le nombre de colonnes est basé sur 12 fractions.

Il y a 6 classes pour s'adapter aux différents type d'écrans. Ces classes sont préfixées par col- puis le "breakpoint" (xs, sm, md, lg, xl, xxl) puis -proportion (de 1 à 12 fractions), exemple:

```
<div class="row">
  <div class="col-12 col-md-6">Hello</div>
  <div class="col-12 col-md-6">World</div>
</div>
```

Voir la doc pour les breakpoints ici : <https://getbootstrap.com/docs/5.0/layout/grid/>

Breakpoints:

- Extra small (xs)
- Small (sm)
- Medium (md)
- Large (lg)
- Extra large (xl)
- Extra extra large (xxl)

Exercice 2 - RESPONSIVE

Soit le fichier HTML suivant possédant 2 row(s).

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <style>
      /* applique le css à toutes les classes dont le nom contient la chaine col- */
      [class*="col-"] {
        padding-top: 15px;
        padding-bottom: 15px;
      }

      .gris { background-color: grey;}

      .jaune { background-color: yellow ; }

      .vert { background-color: green; }
    </style>
  </head>
  <body>
    <div class="container">
      <h3>Mix</h3>
      <div class="row">
        <div class="gris">GRIS </div>
        <div class="vert">VERT </div>
      </div>

      <div class="row">
        <div class="jaune">JAUNE</div>
        <div class="jaune">JAUNE</div>
        <div class="jaune">JAUNE</div>
      </div>
    </div>
  </body>
</html>
```

Exercice 1 :

Créer ce fichier HTML/CSS (pas de serveur web nécessaire).

Installer un lien CDN avec le css Bootstrap (ici le Bootstrap.js n'est pas nécessaire).

Ajouter les attributs grid de bootstrap permettant d'afficher :

1) sur un écran de résolution < 576px

1 seule colonne et 2 rangées pour le 1er row.

1 seule colonne et 3 rangée pour le 2ème row.

2) sur écran de résolution > 576px et < 992px) :

2 colonnes proportion égale pour le 1er row.

3 colonnes de même proportion pour le 2ème row.

3) sur un écran de résolution > 992px

2 colonnes de proportion 2/3 - 1/3 pour le 1er row.

3 colonnes de même proportion pour le 2ème row.

Résultat attendu pour le 3

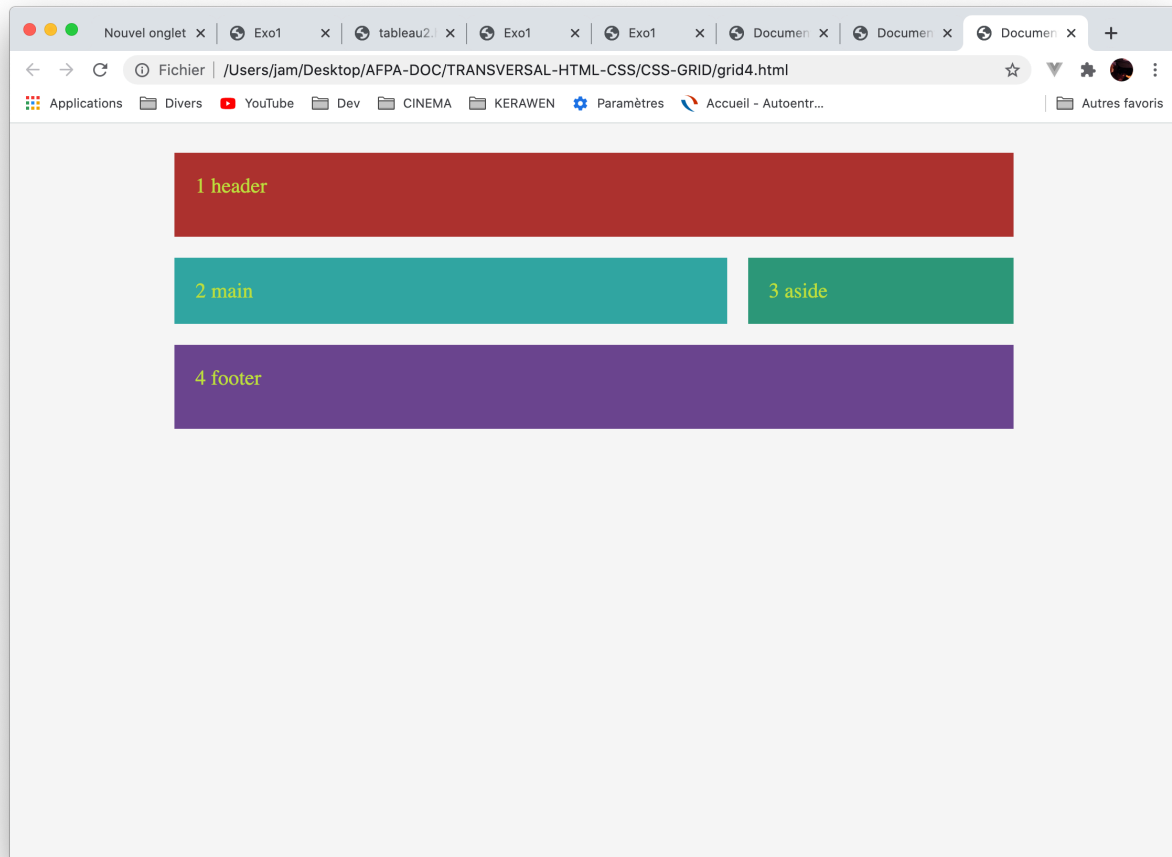
Mix

GRIS		VERT
JAUNE	JAUNE	JAUNE

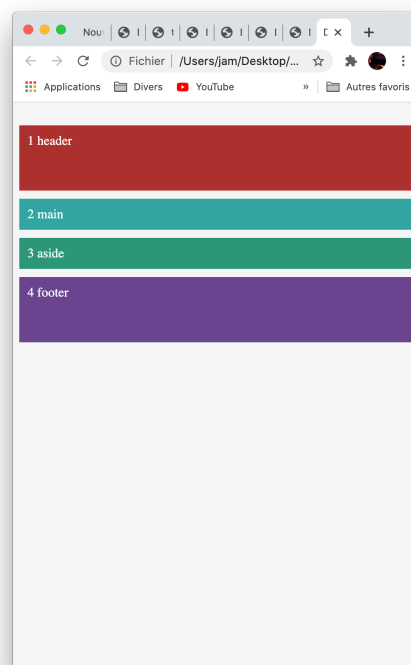
Faire la même chose sans bootstrap et avec CSS GRID ou FLEX

Exercice 3 - GRID et MEDIA-QUERIES

Créer une page qui s'approche de cette configuration de grilles avec CSS GRID



Modifier le css pour que la page s'affiche ainsi quand la résolution est inférieure à 600px (media-queries)



Exercice supplémentaire

Utiliser une police externe avec CSS

Comme pour les vidéos il faut plusieurs types de fichiers pour exploiter les polices sur les différents navigateurs

Pour ça il faut uploader une police choisie sur DAFONT par exemple puis générer le ou les fichiers nécessaires avec un site très utile : <https://www.fontsquirrel.com/tools/webfont-generator>

- Télécharger une police sur Dafont
- la passer dans Fontsquirrel
- télécharger le fichier et décompresser
- On aura des fichiers de type .woff reconnus aujourd'hui
- puis insérer l'exemple CSS fourni

/*! Generated by Font Squirrel (<https://www.fontsquirrel.com>) on August 22, 2020 */

```
@font-face {  
  font-family: 'quadrantaregular';  
  src: url('quadranta-regular-webfont.woff2') format('woff2'),  
       url('quadranta-regular-webfont.woff') format('woff');  
  font-weight: normal;  
  font-style: normal;  
}
```

Puis utiliser cette font-family, exemple:

```
h1 { font-family: 'quadrantaregular' ;}
```