

Sujet

HTML - CSS révisions

TABLE DES MATIÈRES

Nouvelle matière:

- [1. Donner une «class» à une <div>](#)
- [2. Créer une div en plein-écran](#)
- [3. Gérer des colonnes](#)
- [4. Mettre une video en fond](#)
- [5. Créer des ancrs \(liens vers des ID\)](#)
- [6. Réaliser une barre de navigation](#)
- [7. Gérer les positions fixes ou absolues](#)
- [8. Faire des animations à l'aide des transitions](#)

NOUVELLE MATIÈRE

1. Donner une «class» à une <div>

<div> </div>

La balise <div> est un conteneur (un cadre) qui peut inclure tous les tags html (tels que les paragraphes, les listes, les images ..., et également d'autres divisions).

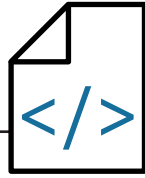
Cet élément est l'un des plus utilisés pour structurer les pages html en plusieurs grandes sections (en tête, partie principale, partie navigation, pied de page ...).

Comme la div va être utilisée pour différentes fonctions, et donc être stylisée sous différentes formes, on ne peut pas l'appeler dans un fichier CSS par le simple sélecteur div{}

Pour pouvoir customiser chaque div individuellement nous allons utiliser une «class» pour lui donner un nom.

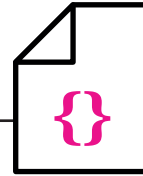
1. Donner une «class» à une <div> (suite)

<div class="nom_de_la_div"><div>



Dans le code HTML :

```
<div class="conteneur"> </div>
```



Dans le code CSS :

```
.conteneur{  
width: 200px;  
height: 100px;  
background-color:# FF6700;  
}
```

Résultat :



Le sélecteur d'une class est toujours précédé par un point.

2. Créer une div en plein-écran

Nous allons réaliser une <div> qui contiendra votre vidéo et qui s'ouvrira en pleine page. Pour l'instant nous lui donnerons une couleur noire.

Exercice

1. Dans votre code HTML, imbriquez dans la balise <body></body> une balise <div></div> et donnez lui une class " upPage"
2. Dans votre feuille de style CSS, recopiez ce code :

```
.upPage{
  background-color: rgba(0,0,0,1.00);
  height:100px;
  width:100%;
}
```

3. Observez le résultat dans chrome, vous obtenez un rectangle noir qui ne touche pas les bords de la page, ce dernier fait 100px de haut et 100% de large
4. Nous allons changer les propriétés pour que cette div s'affiche en plein-écran. Dans votre feuille de style, vous allez changer les propriétés des balises html et body.

```
▼ html{
  height:100%;
}
▼ body{
  height:100%;
  margin:0;
  padding:0;
}
```

Ces propriétés permettent de définir la hauteur des deux balises (100%) et de supprimer les espaces à l'intérieur de la page.

5. Modifiez le code CSS du .conteneur.

```
.upPage{
  background-color: rgba(0,0,0,1.00);
  min-height: 100%;
  width: 100%;
}
```

6. Observez le résultat dans chrome, vous obtenez un rectangle noir qui couvre toute la page et dans lequel nous pourrions mettre votre vidéo.

3. Gérer des colonnes

Nous allons gérer des colonnes qui peuvent être remplies de textes ou d'images. Pour commencer, nous allons diviser la page en 4 colonnes réparties dans un cadre de 1200px.

1. Ajoutez sous la <div> "upPage" une nouvelle <div> nommée "container"

```
<body>
<div class="upPage"></div>
<div class="container"></div>
</body>
```

2. Donnez-lui les propriétés suivantes dans le CSS:

```
.container{
  background-color:#fff;
  overflow:hidden;
  padding-top:20px;
  padding-bottom:20px;
}
```

overflow est une propriété indispensable pour la suite, lorsqu'on utilisera la propriété float. padding permet de vous mettre des espaces dans la balise <div> et ainsi éviter que le contenu (texte, image) ne colle sur les bords. Par défaut votre <div> prend 100% de largeur.

3. Imbriqué une <div class="containerFour"></div> dans la div "container" comme ceci:

```
<body>
<div class="upPage"></div>
<div class="container">
  <div class="containerFour"></div>
</div>
</body>
```

4. Dans le CSS

```
.containerFour{
  width:1200px;
  margin:auto;
  overflow:hidden;
}
```

cette div a une largeur (width) de 1200px. Le margin, est un espace qui se trouve à l'extérieur de la div. Un margin «auto» permet de centrer la div containerFour.

5. Dans cette dernière div nous allons enfin ajouter 4 nouvelles div l'une à la suite de l'autre qui seront nommées "four" et qui représenteront nos 4 colonnes.

6. Dans le code HTML

```
<body>
<div class="upPage"></div>
<div class="container">
  <div class="containerFour">
    <div class="four"></div>
    <div class="four"></div>
    <div class="four"></div>
    <div class="four"></div>
  </div>
</div>
</body>
```

7. Dans le CSS

```
.four{
  width:25%;
  float:left;
}
```

float est une propriété qui place les éléments côte à côte. Chacune des <div> qui aura comme class "four" aura une largeur de 25% par rapport à son conteneur la <div> "containerFour" ce qui nous donnera 4 colonnes

Petit test, de quel % sera la valeur de width, si vous voulez diviser en 2 colonnes ou en 3 colonnes ?

8. Remplissez-les <div class="four"> de texte </div>

9. Testez votre page sur Chrome

10. Dernier petit truc, si vous souhaitez mettre un peu d'espace entre vos textes, voici du code à ajouter dans l'HTML et le CSS

```
<div class="four">
  <div class="content">texte
  de remplissage lorem
  ipsum</div>
</div>
```

```
.content{
  padding:20px;
}
```

4. Mettre une vidéo en fond

Depuis la version 5 de l'HTML, il est très facile d'insérer une vidéo avec la balise `<video></video>`.

Pour cela il vous faudra enregistrer une courte vidéo au format mp4.

1. Dans le dossier de votre site, placez un dossier «video» et mettez-y votre vidéo avec un nom simple suivi de l'extension.

```
<video autoplay loop>
  <source type="video/mp4" src="video/milk.mp4" />
</video>
```

2. Copiez ce code dans l'HTML entre `<div class="upPage">` et `</div>` (pensez à changer le nom milk.mp4 par le nom de votre vidéo)

```
video{
  min-width: 100%;
  min-height: 100%;
  width: auto;
  height: auto;
  opacity:0.5;
  position:fixed;
}
```

3. Dans le CSS les 4 premières propriétés obligent la vidéo à s'afficher en pleine page. "opacity" donne la transparence de l'élément (la valeur va de 1 à 0). "position:fixed" oblige la vidéo à ne pas monter lorsqu'on fait défiler la page.

5. Créer des ancres

Pour ajouter une ancre dans une page et ainsi créer un lien direct dans cette même page, il suffit d'attribuer un "id" à une balise.

4. Dans le code HTML ajoutez une `id="top"` sur la balise `<div class="upPage">`

```
<div class="upPage" id="top">
```

5. ajoutez une `id="section1"` dans la balise `<div class="container" id="center">`

```
<div class="container" id="section1">
```

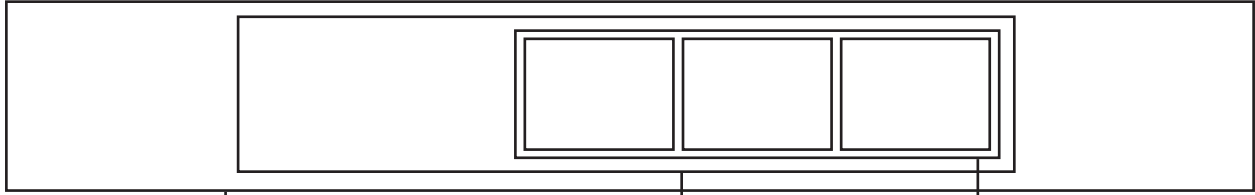
Ces "id" permettront de faire des liens dans la barre de navigation.

Pour info:

les "id" peuvent aussi être utilisés comme sélecteurs CSS. À la différence des "class" qui sont appelées par un point (.) dans les feuilles de style, les "id" sont appelés avec un hashtag (#). Les "id" ne peuvent être utilisés qu'une fois dans une page HTML alors que les "class" peuvent être récupérées plusieurs fois.

6. Réaliser une barre de navigation

Pour réaliser notre navigation, nous aurons besoin d'une série de cadres qui contiendront nos liens.



navContainer, 100% de la fenêtre navigation, 1200px, centrée

Une liste composée de `` et de `` qui contiendront les liens de notre navigation

1. Dans votre code HTML, juste après le tag `<body>` ajoutez ce code :

```
<div class="navContainer">
  <div class="navigation">
    <ul>
      <li><a href="#top"><span>Home</span></a></li>
      <li><a href="#section1"><span>Section</span></a></li>
      <li><a href="#bottom"><span>footer</span></a></li>
    </ul>
  </div>
</div>
```

2. Faites le test dans Chrome. Nous allons transformer cette liste à puce en barre de navigation.
3. Dans la feuille de style CSS, en copiant ce code vous fixez la `<div>` "navContainer" à 100% avec un fond noir. La `<div>` "navigation", sera centrée dans la page et fera 1200px de large.

```
.navContainer{
  width:100%;
  overflow:hidden;
  background-color: rgba(0,0,0,1) ;
}

.navigation{
  width:1200px;
  margin:auto;
}
```

4. Occupons-nous de la liste en supprimant les points (List-style:none) et les retours à la ligne (float:left) :

```
.navigation ul{
  list-style:none;
  padding:0;
  float:right;
  margin:0;
}
.navigation ul li{
  float:left;
  margin:0;
  padding:0;
}
```

Il existe plusieurs sortes de sélecteurs CSS. Nous avons déjà vu le sélecteur avec une balise, avec une class et un id. Vous remarquerez qu'ici, qu'on fait appel à une suite dite «descendant». Voir les différents sélecteurs sur ce site : <https://code.tutsplus.com/fr/tutorials/the-30-css-selectors-you-must-memorize--net-16048>

5. Les liens <a> ne peuvent pas avoir de taille fixe (height,width) pour contourner ce problème il faut utiliser la propriété "display:block;"

```
.navigation ul li a{
  color:#fff;
  text-align:center;
  text-decoration:none;
  display:block;
  padding:0;
}
.navigation ul li a span{
  padding:40px 60px;
  display:block;
}
```


7. Gérer les positions fixes ou absolues

La propriété "position" permet, soit de fixer (valeur "fixed") un élément dans une zone du navigateur, soit de faire passer un élément au-dessus d'un autre avec la valeur "absolute"

1. Dans le CSS ajoutez au sélecteur ".navContainer" la propriété "position:fixed"

```
.navContainer{
  position:fixed;
  z-index:1000;
  top:0;
  width:100%;
  overflow:hidden;
  background-color: rgba(0,0,0,1) ;
}
```

La propriété "z-index:1000;" met la <div class="navContainer"> au-dessus des autres

8. Faire des animations à l'aide des transitions

Les transitions sont des propriétés CSS qui permettent de réaliser de courtes animations.

Pour commencer, nous allons faire réagir les liens de notre barre de navigation lorsque le curseur survolera ces derniers.

1. Dans le code CSS vous allez ajouter ceci:

```
.navigation ul li a:hover{}
```

le :hover est une «pseudo-classe» qui a pour rôle d'appliquer les propriétés au survol de la balise. C'est à dire, que les propriétés qui seront entre les deux accolades {} ne seront prises en compte que lors du survol du curseur.

2. Entre les accolades { } ajoutez ces propriétés :

```
.navigation ul li a:hover{
  background-color: rgba(255,255,255,1.00);
  color: rgba(0,0,0,1.00);
}
```

3. Faites le test dans Chrome.
Le changement est assez net, nous allons le rendre plus doux grâce aux transitions.
4. Remontez au sélecteur suivant, **ne le réécrivez pas, il est dans votre code CSS** :

```
.navigation ul li a{
  color:#fff;
  text-align:center;
  text-decoration:none;
  display:block;
  padding:0;
}
```

5. Ajoutez après «padding:0;» ce code-ci:

```
transition-property: all;  
transition-duration: 1s;  
transition-timing-function: ease-in;  
transition-delay: 0.1s;
```

Vous pouvez aussi résumer ces quatre lignes de code en une seule:
transition: all 1s ease-in 0.1s;

6. Testez votre page dans Chrome

Si vous voulez aller plus loin avec les transitions voici trois liens intéressants:

<http://www.alsacreations.com/tuto/lire/873-transitions-css3-animations.html>

<https://openclassrooms.com/courses/utilisez-les-effets-avances-de-css-sur-votre-site/les-transitions-css-1>

https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Transitions/Utiliser_transitions_CSS