

TD2 CSS : HTML/CSS Avancé Partie 2

SOMMAIRE

Exercice 1 :

Exercice 2 :

Exercice 3 :

Exercice 4 :

Exercice 5 :

Exercice 6 :

Exercice 7 :

Exercice 8 :

Exercice 9 :

Exercice 10 :

Exercice 11 :

Exercice 12 :

Exercice 1 :

#id → (1, 1, 0, 0)

nav.titi .tata div div div div div → (1, 0, 2, 6)

ul li div.skill → (1, 0, 1, 3)

.titi span → (1, 0, 1, 1)

div span → (1, 0, 0, 2)

div > a → (1, 0, 0, 2)

div + a → (1, 0, 0, 2)

Exercice 2 :

Exercice 2

Quelle est la couleur du texte “Priorité CSS” dans les deux cas suivant ? Quelle règle de priorité CSS explique votre réponse ?

1. Le fichier **styles.css** contient `p {color:blue;}`, et le fichier **index.html** contient

```
<html>
<head>
  <style type="text/css">
    p {color: pink}
  </style>
  <link rel="stylesheet" type="text/css" href="css/styles.css">
</head>
<body><p style="color:red">Priorité CSS</p></body>
</html>
```

2. Le même fichier **styles.css** et le fichier **index.html** sans la règle inline

```
<html>
<head>
  <style type="text/css">
    p {color: pink}
  </style>
  <link rel="stylesheet" type="text/css" href="css/styles.css">
</head>
<body><p>Priorité CSS</p></body>
</html>
```

1. Dans le premier cas, la couleur du texte “Priorité CSS” est la couleur rouge car la règle CSS définie en style inline (style="color:red") a la spécificité la plus élevée et est prioritaire sur les styles internes et externes.
2. Dans le deuxième cas la couleur du texte “Priorité CSS” est la couleur rose car il n’y a plus de style inline ; les deux règles p {} (interne et externe) ont la

même spécificité, et c'est la dernière chargée dans le document (la règle interne placée après l'externe) qui l'emporte.

Exercice 3 :

Pour commencer dans l'exercice 3 je vais commencer par changer le menu de mon site par le suivant :

```
<nav>
  <div><a href="./index.html">Accueil</a></div>
  <div><a href="./facts.html">Facts</a></div>
  <div><a href="./news.html">Actualités</a></div>
  <div><a href="./contact.html">Contact</a></div>
</nav>
<main>
```

Ensuite pour la prochaine étape je vais donner une nouvelle largeur au nav de 75% et lui ajouter des marges horizontales en auto :

```
nav {
  width: 75%;
  margin: 0 auto;
}
```

Ensuite je vais ajouter des règles css afin que les fils <div> enfants de <nav> aient une couleur de fond :

```
nav > div {
  background-color: #5BBDBF;
}
```

Et pour finir l'exercice 3 je vais ajouter une règle CSS pour que les éléments <a> descendants de <nav> aient une couleur de fond :

```
nav a {
  background-color: #c0d5c2;
}
```

Exercice 4 :

Pour l'exercice 4 je vais commencer par donner aux `<div>` enfants de `<nav>` le `display inline`:

```
✓ nav > div {  
  display: inline;  
}
```

Ensuite je vais aux `<nav>` une hauteur de 50px :

```
nav {  
  height: 50px;  
}
```

Puis je vais ajouter du padding horizontal de 10px sur les éléments `<a>` :

```
nav a {  
  padding: 0 10px;  
}
```

Je vais ensuite ajouter a ces même éléments des une bordure sur la gauche de 2px de style solid et de couleur noire :

```
✓ nav a {  
  padding: 0 10px;  
  border-left: 2px solid #000;  
}
```

Exercice 5 :

Pour l'exercice 5 je vais ajouter des sous menu :

```
<nav>
  <div><a href="./index.html">Accueil</a><div class="submenu">
    <div><a href="./one.html">one</a></div>
    <div><a href="./two.html">two</a></div>
    <div><a href="./three.html">three</a></div>
  </div></div>

  <div><a href="./facts.html">Facts</a></div>
  <div><a href="./news.html">Actualités</a></div>
  <div><a href="./contact.html">Contact</a><div class="submenu">
    <div><a href="./other.html">other</a></div>
    <div><a href="./another.html">another</a></div>
  </div></div>
```

Ensuite nous souhaitons que l'affichage du reste de la page fasse comme si ces sous menu n'existaient pas, pour ce faire nous allons effectuer ça en plusieurs étapes, donc pour la première étape je vais créer la règle CSS qui affecte cette valeur de position aux balises de classe submenu avec un décalage de 50px par rapport au haut et de 0px par rapport à la gauche :

```
.submenu {
  position: absolute;
  top: 50px;
  left: 0px;
}
```

Exercice 6 :

On va commencer par donner au sous menus la couleur de fond #aca :

```
background-color: #aca;  
display: none;  
}
```

Pour les réafficher quand on les survol il faut faire ça :

```
nav > div:hover .submenu{  
    display: block;  
}
```

Exercice 7 :

Pour commencer on va donner à l'élément <nav> la valeur display flex :

```
nav {  
    height: 50px;  
    display: flex;  
}
```

Ensuite on va changer la valeur des display <div> inline en block :

```
nav > div {  
    display: block;  
    position: relative;  
}
```

Ensuite on va donner une largeur fixe de 100px à tous les div descendants de nav :

```
nav > div {  
  display: block;  
  position: relative;  
  width: 100px;  
}
```

Exercice 8 :

Pour commencer on va aligner les éléments enfants de nav à l'aide de la propriété align item:

```
nav a {  
  background-color: #c0d5c2;  
  display: flex;  
  height: 50px;  
  align-items: center;  
}
```

Exercice 9 :

Pour cet exercice, on devait placer les titres du menu à droite dans la barre de navigation <nav> et vérifier visuellement le résultat.

```
✓ nav {  
  height: 50px;  
  display: flex;  
  justify-content: flex-end;  
  align-items: center;  
  background: #ffdede;  
}
```



Exercice 10 :

- 1) pour commencer on va placer le menu sur la même ligne que la citation :

```
✓ header {  
    display: flex;  
}
```

- 2) Ensuite on va positionner le menu pour qu'il soit en bas de la balise :

```
header {  
    display: flex; /*display*/  
    align-items: flex-end;  
}
```

- 3) Pour mettre le fond blanc il suffit de mettre cette ligne :

```
background-color: white;
```


Exercice 11 :

On va donner le au body le width 900px :

```
body{
    background-color: #838892;
    width: 900px;
    margin: auto;
}
```

On va ajouter display flex sur le main :

```
main {  
  display: flex;  
}
```

Résultat :

[illegible]

3) on va fixer la largeur de article et aside et ajouter une marge de 10% a gauche :

```
✓ article{  
  width: 60%;  
  margin-left: 10%;  
}  
  
✓ aside{  
  width: 30%;  
  margin-left: 10%;  
}
```

4) Et pour finir on va mettre le background color de aside et article :

```
✓ article{  
  width: 60%;  
  background-color: #ccc;  
}  
  
✓ aside{  
  width: 30%;  
  margin-left: 10%;  
  background-color: #ccc;  
}
```

Exercice 12 :

- 1) Pour ajouter les puces il suffit d'ajouter des `` :

```
<nav> <!--Balises navigation-->
<div><a style="background-color: #5bbdbf;" href="/index.html">Accueil</a>
  <div class="submenu">
    <div><a href="/one.html"><span class="puce">■</span>one</a></div>
    <div><a href="/two.html"><span class="puce">■</span>two</a></div>
    <div><a href="/three.html"><span class="puce">■</span>three</a></div>
  </div> <!--div contenant balise a-->
<!--><div><a style="background-color: #5bbdbf;" href="/facts.html">Facts</a></div> <!--div contenant balise a-->
<!--><div><a style="background-color: #5bbdbf;" href="/news.html">Actualités</a></div> <!--div contenant balise a-->
<!--><div><a style="background-color: #5bbdbf;" href="/contact.html">Contact</a>
  <div class="submenu">
    <div><a href="/other.html"><span class="puce">■</span>other</a></div>
    <div><a href="/another.html"><span class="puce">■</span>another</a></div>
```

- 2) on va ensuite ajouter grâce à la class puce et faire un `.puce` avec `visibility: hidden`;

```
nav > div .submenu > div:hover .puce{
  visibility: visible;
  color: black;
```