

Barre de navigation CSS

Avoir une navigation facile à utiliser est important pour tout site Web.

Avec CSS, vous pouvez transformer des menus HTML ennuyeux en barres de navigation attrayantes.

Barre de navigation = Liste des liens

Une barre de navigation a besoin de HTML standard comme base.

Dans nos exemples, nous allons construire la barre de navigation à partir d'une liste HTML standard.

Une barre de navigation est essentiellement une liste de liens, donc l'utilisation des éléments `` et `` est parfaitement logique :

```
<ul>
  <li><a href="#home">Accueil</a></li>
  <li><a href="#news">Service</a></li>
  <li><a href="#contact">A propos</a></li>
  <li><a href="#about">Contact</a></li>
</ul>
```

Supprimons maintenant les puces, les marges et le padding de la liste :

```
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
}
```

Exemple expliqué

- `list-style-type: none;` - Supprime les puces. Une barre de navigation n'a pas besoin de marqueurs de liste
- Définir `margin: 0;` et `padding: 0;` - Supprimer les paramètres par défaut du navigateur

Le code de l'exemple ci-dessus est le code standard utilisé dans les barres de navigation verticales et horizontales, sur lequel vous en apprendrez plus dans les slides suivants.

Barre de navigation verticale CSS

Pour créer une barre de navigation verticale, vous pouvez styliser les éléments `<a>` à l'intérieur de la liste, en plus du code de la page précédente :

Exemple expliqué

- `display: block;` - L'affichage des liens sous forme d'éléments de bloc rend toute la zone de lien cliquable (pas seulement le texte), et cela nous permet de spécifier la largeur (le padding, la marge, la hauteur, etc. si vous le souhaitez)
- `width: 60px;` - Les éléments de bloc occupent toute la largeur disponible par défaut. Nous voulons spécifier une largeur de 60 pixels

Vous pouvez également définir la largeur de `` et supprimer la largeur de `<a>`, car ils occuperont toute la largeur disponible lorsqu'ils seront affichés en tant qu'éléments de bloc. Cela produira le même résultat que notre exemple précédent :

Exemples de barre de navigation verticale

Créez une barre de navigation verticale de base avec une couleur d'arrière-plan grise et modifiez la couleur d'arrière-plan des liens lorsque l'utilisateur passe la souris dessus :

```
li a:hover {  
    background-color: #555;  
    color: white;  
}
```

Lien de navigation actif/actuel

Ajoutez une classe "active" au lien actuel pour indiquer à l'utilisateur sur quelle page il se trouve :

```
li a.active {  
  background-color: #04AA6D;  
  color: white;  
}
```

Barre de navigation verticale fixe pleine hauteur

Créez une navigation latérale "collante" pleine hauteur :

```
ul {  
  list-style-type: none;  
  margin: 0;  
  padding: 0;  
  width: 25%;  
  background-color: #f1f1f1;  
  position: fixed;  
  height: 100%;  
  overflow: auto;  
}
```

Remarque : Cet exemple peut ne pas fonctionner correctement sur les appareils mobiles.

Barre de navigation horizontale CSS

Il existe deux manières de créer une barre de navigation horizontale. Utilisation d'éléments de liste en **ligne** ou **flottants**

Éléments de liste en ligne

Une façon de créer une barre de navigation horizontale consiste à spécifier les éléments `` comme `inline`, en plus du code "standard" de la page précédente :

```
li {  
  display: inline;  
}
```

Exemple expliqué :

`display: inline;` - Par défaut, les éléments `` sont des éléments de bloc. Ici, nous supprimons les sauts de ligne avant et après chaque élément de la liste, pour les afficher sur une seule ligne

Éléments de la liste flottante

Une autre façon de créer une barre de navigation horizontale consiste à faire flotter les éléments `` et à spécifier une disposition pour les liens de navigation :

```
li {  
  float: left;  
}
```

Exemple expliqué :

- `float: left;` - Utilisez `float` pour faire flotter les éléments de bloc les uns à côté des autres
- `display: block;` - Nous permet de spécifier le padding (et la hauteur, la largeur, les marges, etc. si vous le souhaitez)
- `padding: 8px;` - Spécifiez un padding entre chaque élément `<a>`, pour leur donner une belle apparence
- `background-color: #dddddd;` - Ajouter une couleur de fond grise à chaque élément `<a>`

Astuce : Ajoutez la couleur d'arrière-plan à `` au lieu de chaque élément `<a>` si vous voulez une couleur d'arrière-plan pleine largeur :

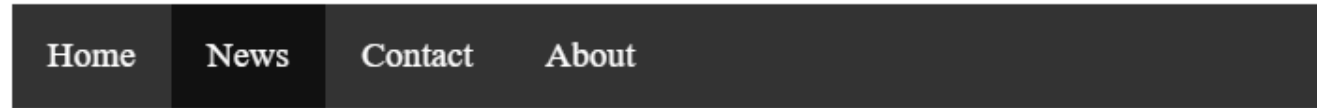
```
ul {  
    background-color: #dddddd;  
}
```

```
li {  
    float: left;  
}
```

Exercice de barre de navigation horizontale

Créez une barre de navigation horizontale de base avec une couleur d'arrière-plan sombre (noir clair) et modifiez la couleur d'arrière-plan (noir foncé) des liens lorsque l'utilisateur passe la souris dessus :

Exemple en image:



Ajoutez une classe "active" au lien actuel pour indiquer à l'utilisateur sur quelle page il se trouve :

Exemple en image:



Liens alignés à droite

On peut aligner les liens à droite en faisant flotter les éléments de la liste vers la droite (float:right;) :

```
<ul>
  <li><a href="#home">Accueil</a></li>
  <li><a href="#news">Services</a></li>
  <li><a href="#contact">Contact</a></li>
  <li style="float:right"><a class="active" href="#about">À propos</a></li>
</ul>
```

Barre de navigation fixe

Faites en sorte que la barre de navigation reste en haut ou en bas de la page, même lorsque l'utilisateur fait défiler la page :

Navigation fixe en haut

```
ul {  
  list-style-type: none;  
  margin: 0;  
  padding: 0;  
  overflow: hidden;  
  background-color: #333;  
  position: fixed;  
  top: 0;  
  width: 100%;  
}
```

Navigation fixe en bas

```
ul {  
  list-style-type: none;  
  margin: 0;  
  padding: 0;  
  overflow: hidden;  
  background-color: #333;  
  position: fixed;  
  bottom: 0;  
  width: 100%;  
}
```

Barre de navigation collante

Ajoutez `position: sticky;` à `` pour créer une barre de navigation collante.

Un élément collant bascule entre relatif et fixe, en fonction de la position de défilement. Il est positionné de manière relative jusqu'à ce qu'une position de décalage donnée soit rencontrée dans la fenêtre - puis il "colle" en place (comme `position:fixed`).

```
ul {  
  list-style-type: none;  
  margin: 0;  
  padding: 0;  
  overflow: hidden;  
  background-color: #333;  
  position: -webkit-sticky; /* Safari */  
  position: sticky;  
  top: 0;  
}
```