

6. SIGNALER LA PAGE COURANTE DANS LA BARRE DE NAVIGATION

Pour les petits sites où la navigation est incorporée dans la page, vous pouvez ajouter simplement la classe dans chaque page mais pour la majorité des sites, il reste plus intéressant de signaler la page sur laquelle on se trouve sans avoir à ajouter une classe au menu.

Cette technique fonctionne (dans des sites statiques) par l'ajout d'un ID ou d'un nom de classe à l'élément body de chaque page (indiquant dans quelle page ou quelle section se trouve l'utilisateur). Ajoutez ensuite un ID ou un nom de classe correspondant à chaque élément de la liste de navigation (souvent construit à partir du nom de la page à laquelle est lié l'élément de la liste) :

```
<body id="home">
<ul class="nav">
    <li class="home"><a href="home.htm">Accueil</a></li>
    <li class="about"><a href="about.htm">À propos</a></li>
    <li class="services"><a href="services.htm">Nos services</a></li>
    <li class="products"><a href="work.htm">Notre travail</a></li>
    <li class="news"><a href="news.htm">Actualités</a></li>
    <li class="contact"><a href="contact.htm">Contact</a></li>
</ul>
</body>
```

Pour signaler ensuite la page courante en CSS :

```
#home .nav .home a,
#about .nav .about a,
#news .nav .news a,
#products .nav .products a,
#services .nav .services a {
    background-position: right bottom;
    color: #fff;
    cursor: default;
/* le curseur ne changeant pas d'état, l'utilisateur n'est pas tenté de cliquer
sur le lien vers la page où il se trouve déjà */
}
```

Cette méthode consiste à utiliser des sélecteurs descendants pour contrôler la mise en forme de la navigation principale en fonction de l'id de l'élément body de chaque page.

La première partie du sélecteur home vérifie que nous sommes sur cette page, la mise en forme n'est appliquée que si l'élément a est un enfant de home qui lui-même doit être un enfant de nav. Si une correspondance est trouvée, l'action est réalisée.

7. MENU DEROULANT DISPLAY

Cette technique consiste à cacher les éléments de niveau 2 (ou plus) en leur appliquant un display: none, et à les restaurer au survol de leur parent via un display: block.

```
.sub-nav {
    display: none; /* Masque la sous-liste */
}
.nav-item:hover .sub-nav {
    display: block; /* Affiche cette sous-liste au survol de son conteneur */
}
```

L'inconvénient de cette méthode est que les éléments n'existent plus dans le document : il est impossible d'y accéder en tabulant, et les synthèses vocales les ignoreront royalement.

8. MENU DEROULANT SUCKERFISH¹

La meilleure solution pour masquer visuellement un élément, tout en le gardant accessible pour les aides techniques (synthèses vocales, etc.) et quand on navigue au clavier est de le sortir de l'écran via un positionnement absolu et de le faire réapparaître ensuite en rétablissant son positionnement. Les liens qu'elles contiennent sont alors accessibles en tabulant.

```
ul nav li ul {  
    position: absolute;  
    width: 8em;  
    /*Pour être sûr que les éléments dans les menus déroulants s'empilent  
    verticalement, il faut définir la largeur de la liste en lui attribuant la même  
    valeur que la largeur des éléments de liste incorporés */  
    left: -999em;  
    /*Pour masquer les menus déroulants jusqu'à ce qu'ils soient activés, il faut les  
    positionner de manière absolue (position: absolute) en les sortant à gauche de  
    l'écran */  
    margin-left: -1px;  
}  
  
nav li:hover ul {  
    left: auto;  
    /*En ajoutant une pseudo-classe hover à l'élément de liste parent, on fait  
    réapparaître la liste en ramenant sa position à son emplacement de départ */  
}
```

Cette technique fonctionne dans la plupart des navigateurs web modernes, sauf les anciennes versions d'Internet Explorer, qui ne reconnaissent pas la pseudo-classe :hover pour les éléments qui ne sont pas des ancrages. Pour contourner ce problème, utilisez le JavaScript permettant d'activer cette fonctionnalité (plus de détails à ce sujet <http://htmldog.com/articles/suckerfish/dropdowns/> - en anglais).

Lecture Web :

<http://css.maxdesign.com.au/listamatic/> site web en anglais entièrement consacré à la conception de menus en CSS, vous y trouverez de nombreuses versions de menus verticaux, horizontaux, déroulants, et d'autres plus originaux encore.

<http://css.alsacreations.com/Galeries-de-menus-en-CSS> Alsacréations propose une galerie de menus en CSS, principalement graphiques.

9. DES MENUS DEROULANTS ACCESSIBLES

Pour être accessible un menu déroulant doit respecter trois grands principes :

1. Pouvoir être restitué par une synthèse vocale (donc pas de display: none) ;
2. Pouvoir être utilisé au clavier (focus) en conservant un ordre logique ;
3. Pouvoir être utilisé même lorsque JavaScript n'est pas disponible (état déplié par défaut).

Plus d'infos

<http://www.fairytells.net/index.php/2006/07/22/35-la-recette-pour-un-menu-deroulant-accessible-partie-2>

http://openweb.eu.org/articles/menu_universel

<http://www.hightdots.com/products/css-tab-designer/> Créeur de menu

¹ Elaborée par Patrick Griffith -<http://www.alistapart.com/articles/dropdowns/>