

TP : Réaliser un carrousel

Votre travail consiste à réaliser un carrousel grâce à jQuery.

Explications et pré-requis

Un carrousel, c'est tout simplement un défilement d'images, le plus souvent présent sur l'index d'un site web. Cela permet de gagner de la place, tout en restant très esthétique, et faire une présentation originale. Nous vous invitons à aller voir quelques exemples de carrousel plus ou moins sophistiqués aux adresses suivantes :

- [Grafikart.fr \(index\)](http://Grafikart.fr) ;
- [CodeCanyon.net \(jQuery Banner Rotator\)](http://CodeCanyon.net) ;
- Wowslider.com ;

Objectif concret

Vous l'aurez maintenant compris, votre travail va être de réaliser vous-même un carrousel. Simple dans un premier temps, vous pourrez toujours l'améliorer par la suite si vous êtes motivé ! Ce carrousel, qu'on appellera également slider assez couramment, devra simplement comporter quelques fonctions basiques :

- Il devra posséder deux boutons, un pour afficher l'image précédente, l'autre pour afficher la suivante ;
- Il saura défiler seul, c'est-à-dire que les images défileront d'elles-mêmes sans que l'on ait à intervenir ;
- Il pourra être très flexible dans la mesure où il sera possible de mettre autant d'images que l'on souhaite.

Tout ce dont vous avez besoin pour réaliser cela, c'est de la patience, et les connaissances que vous venez d'engranger !

Méthodes requises

Vous allez devoir piocher dans chaque chapitre pour pouvoir mener à bien votre travail. En effet, ce TP est complet dans le sens où il utilise tous les domaines de jQuery : les événements, les sélecteurs, les fonctions de parcours, la manipulation du DOM... Vous allez devoir tout mettre à profit, et c'est exactement ce que nous voulons.

Vous devrez donc partir d'un squelette en HTML, très simple, puisqu'il ne s'agit en fait que d'un bloc contenant une liste d'images. Rien d'extraordinaire, et le CSS n'a rien à lui envier, car il sera lui aussi très très simple, sauf deux subtilités à ne pas manquer. En effet, l'astuce est de définir une taille fixe à votre carrousel, les images étant empilées les unes sur les autres grâce à une position absolue.

Une fois ceci terminé, tout se passe ensuite du côté de jQuery. L'astuce est de jouer sur la présence des images, grâce à la propriété `display` de CSS. N'hésitez pas non plus à manipuler les index, car c'est comme cela que vous pourrez déterminer quelle image doit être affichée. Enfin, les boutons `précédent` et `suivant` devront être ajoutés à la volée, pour empêcher l'utilisateur ayant désactivé JavaScript de les voir et de cliquer dessus sans voir de résultat.

`setTimeout()`, ou répéter une fonction régulièrement

Une méthode native de JavaScript va vous être fabuleusement utile pour faire défiler automatiquement les images. Il s'agit de `setTimeout()`, qui prend deux arguments : le nom de la fonction à exécuter, et l'intervalle de temps (en millisecondes) à attendre avant de le faire. L'astuce, c'est de relancer la fonction à l'intérieur d'elle-même pour réaliser une boucle infinie :

```
function maBoucle(){
    setTimeout(function(){
        alert('Bonjour !'); // affichera "Bonjour !" toutes les
secondes
        maBoucle(); // relance la fonction
    }, 1000);
}
maBoucle(); // on oublie pas de lancer la fonction une première
fois
```

Si vous souhaitez partir sur de bonnes bases, nous vous suggérons de prendre notre structure HTML et le style CSS minimal :

```
<div id="carrousel">
  <ul>
    <li></li>
    <li></li>
    <li></li>
  </ul>
</div>
```

Les images utilisées ici proviennent de lorempixel.com et ne sont pas toutes libres de droit.

```
#carrousel{
  position:relative;
  height:200px;
  width:700px;
  margin:auto;
}
```

```
#carrousel ul li{
  position:absolute;
  top:0;
  left:0;
}
```

Vous avez maintenant toutes les clés pour réaliser ce carrousel ! N'oubliez pas de consulter la [documentation](#) de jQuery si vous en avez besoin, et de faire usage de Firebug, par exemple, pour localiser un soucis dans votre code. Si vous avez vraiment du mal, n'hésitez pas à poser votre question sur le forum JavaScript !