

A. Les objectifs

Dans ce chapitre nous allons aborder l'API de jQuery. Une **API**, pour *Application Programming Interface*, est constituée d'un ensemble de fonctions, de classes et de méthodes qui permettent de créer des applications. Dans notre cas ce sont des applications web. jQuery est une bibliothèque qui propose une API très bien documentée.

Voici l'URL de l'API de jQuery : <http://api.jquery.com>.

jQuery étant écrit en JavaScript, nous allons voir brièvement quelques éléments de la syntaxe du JavaScript afin d'aborder celle de jQuery plus facilement.

Suite à cette très courte introduction, nous aborderons la syntaxe de jQuery par l'intermédiaire d'un exemple concret pour plus de facilité.

Puis, nous allons apprendre à utiliser jQuery avec ses sélecteurs, avec la manipulation des propriétés CSS et nous apprendrons à gérer les événements et les effets jQuery. L'objectif n'est pas de devenir un développeur expert en jQuery, mais d'avoir suffisamment d'informations pour être à même de modifier et de personnaliser des scripts jQuery que vous pourrez trouver sur le Web.

B. Le JavaScript

La bibliothèque jQuery est écrite en JavaScript. Nous allons donc très rapidement aborder la syntaxe du JavaScript avec un exemple ultrasimple.

Le JavaScript est un langage de programmation de scripts qui va vous permettre de dynamiser vos pages web, de les rendre plus attractives et plus interactives pour les visiteurs. Un point important qu'il faut signaler est que le JavaScript est exécuté dans le navigateur de l'internaute, il n'est pas exécuté sur les serveurs web des hébergeurs. De plus le JavaScript est reconnu par tous les navigateurs web, quelle que soit la plateforme. Il n'y a rien à ajouter dans votre navigateur pour que le JavaScript fonctionne (sauf, bien sûr, si un internaute a désactivé le JavaScript dans son navigateur).

Voyons un premier exemple de script JavaScript :

```
<!doctype html>
<html lang="fr">
<head>
<meta charset="utf-8">
<title>JavaScript</title>
<script>
    alert("Bonjour tout le monde !");
</script>
</head>
<body>
</body>
</html>
```

Détaillons cette page :

- Vous remarquez que cette page n'a pas de contenu dans l'élément `<body>`. Ce n'est absolument pas obligatoire.
- Le code JavaScript est placé dans l'élément HTML `<script>`.
- Le script est composé d'une simple ligne de code, une instruction. Cette instruction utilise une seule fonction : `alert()`. Cette fonction possède un nom qui lui est strictement réservé et une paire de parenthèses afin de pouvoir y placer des paramètres. Cette fonction a pour but d'afficher une boîte de dialogue dans la fenêtre du navigateur, avec un texte.
- Le paramètre de cette fonction est placé entre guillemets car nous souhaitons afficher du texte : `"Bonjour tout le monde !"`.
- Cette instruction se termine par le caractère point-virgule : `;`. Cela permet d'indiquer que l'instruction est terminée.

Voici l'affichage obtenu :



Notez bien que l'aspect de la boîte de dialogue peut varier d'un navigateur à l'autre.

Le fichier à télécharger est **C03-B1.html**.

C. La syntaxe de jQuery

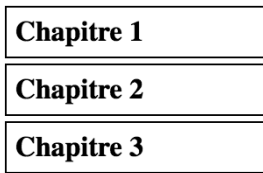
1. Un exemple simple

Plutôt que de rédiger un chapitre austère sur la syntaxe de jQuery, nous allons partir d'un exemple simple et concret. jQuery utilise la syntaxe JavaScript, mais il la simplifie et la rend plus courte et plus rapide à saisir. Notez que dans cette optique de rapidité, jQuery utilise parfois une syntaxe qui lui est propre, pour gagner sur le nombre de lignes de code.

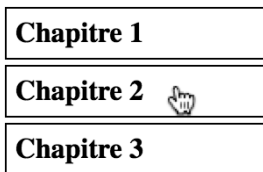
2. L'affichage de la page

Pour bien appréhender cette syntaxe, nous allons donc créer un premier script jQuery très simple.

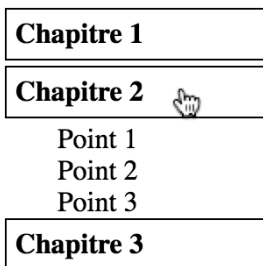
Au chargement de la page, nous affichons des titres de chapitre :



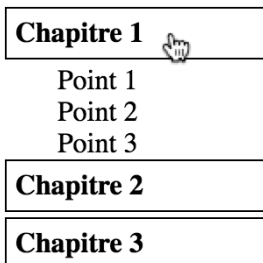
Au survol d'un chapitre, le curseur prend l'aspect d'un "pointeur-main" :



Au clic de la souris, le chapitre se déploie avec un effet de glissement :



Au clic sur un autre chapitre, le chapitre précédent se ferme et celui cliqué s'ouvre, toujours avec des effets de glissement :



3. Le code de la page

Maintenant, voyons le code utilisé :

```
<!doctype html>
<html lang="fr">
<head>
<meta charset="utf-8">
<title>jQuery</title>
<script src="js/jquery.js"></script>
<script>
$(document).ready(function(){
    $("div.point").hide();
    $("p.chapitre").click(function(){
        $(this).next("div.point").slideDown(300)
        .siblings("div.point").slideUp("slow");
    });
});
</script>
<style>
.menu {
    width: 150px;
}
.chapitre {
    padding: 5px;
    cursor: pointer;
    position: relative;
    margin: 0 0 3px 0;
    font-weight: bold;
    border: 1px solid black;
}
.point a {
    display: block;
    color: black;
    background-color: white;
    padding-left: 30px;
    text-decoration: none;
}
.point a:hover {
    color: black;
    text-decoration: underline;
}
}
```

```
</style>
</head>
<body>
<div>
  <div class="menu">
    <p class="chapitre">Chapitre 1</p>
    <div class="point">
      <a href="#">Point 1</a>
      <a href="#">Point 2</a>
      <a href="#">Point 3</a>
    </div>
    <p class="chapitre">Chapitre 2</p>
    <div class="point">
      <a href="#">Point 1</a>
      <a href="#">Point 2</a>
      <a href="#">Point 3</a>
    </div>
    <p class="chapitre">Chapitre 3</p>
    <div class="point">
      <a href="#">Point 1</a>
      <a href="#">Point 2</a>
      <a href="#">Point 3</a>
    </div>
  </div>
</div>
</body>
</html>
```

Le fichier à télécharger est **C03-C3.html**.

4. Le code HTML

Dans un premier temps, voyons la structure de la page :

- L'ensemble de la page est inséré dans une boîte `<div>` anonyme.
- Le menu est placé dans une boîte `<div class="menu">`, utilisant la classe `.menu`.
- Le titre de chaque chapitre est inséré dans un élément `<p>`, avec la classe `.chapitre` :
`<p class="chapitre">Chapitre 1</p>`.
- Chaque chapitre possède des "points" qui sont insérés dans une boîte `<div class="point">`.
- Chaque point est placé dans un lien `<a>`.

5. Le code CSS

La mise en forme se fait avec des règles CSS.

Le menu, avec la classe `.menu` possède une largeur d'affichage de 150 pixels :

```
.menu {  
    width: 150px;  
}
```

Chaque titre de chapitre, avec la classe `.chapitre`, est en position relative, avec des remplissages et des marges définis. Le curseur est un pointeur, le texte est en gras et le titre possède une bordure.

```
.chapitre {  
    padding: 5px;  
    cursor: pointer;  
    position: relative;  
    margin: 0 0 3px 0;  
    font-weight: bold;  
    border: 1px solid black;  
}
```

Chaque point de chapitre, avec le sélecteur `.point a`, s'affiche en mode bloc, doté d'arrière-plan blanc, avec un remplissage, sans décoration de texte doté d'une couleur noire.

```
.point a {  
    display: block;  
    color: black;  
    background-color: white;  
    padding-left: 30px;  
    text-decoration: none;  
}
```

Au survol des points, avec le sélecteur `.point a:hover`, nous avons une décoration de texte en souligné, toujours de couleur noire.

```
.point a:hover {  
    color: black;  
    text-decoration: underline;  
}
```

6. Le script jQuery

Maintenant nous allons aborder la syntaxe du script jQuery.

La bibliothèque jQuery est liée localement au fichier qui a été précédemment téléchargé (reportez-vous au chapitre précédent pour avoir plus de précision) :

```
<script src="js/jquery.js"></script>
```