

jQuery

Table des matières

| | | |
|----|-------------------|---|
| 1. | Introduction..... | 2 |
| 2. | Syntaxe | 2 |

1. Introduction

jQuery est une librairie, plus ou moins comme Bootstrap, qui permet d'écrire du code JavaScript de manière plus rapide / pratique. Il consiste en une série de fonctions pré-écrites que nous allons pouvoir utiliser telles-elles, et possède sa propre syntaxe. Par exemple :

Sélectionner un élément HTML :

```
<div id="maDiv">
  Bonjour
</div>
<script src="https://code.jquery.com/jquery-3.7.1.js" integ
<script>
  // Javascript ici :)
  document.getElementById('maDiv').innerHTML = 'Salut';
  // jQuery ici :)
  $('#maDiv').html('Salut !')
</script>
```

Il fonctionne sur une logique de **sélecteurs**, qui sont les mêmes qu'en CSS, c'est-à-dire les balises html, le point **.** pour les classes et le dièse **#** pour les id. Et pour l'intégrer on utilise comme avec Bootstrap, un **CDN**.

<https://releases.jquery.com/>

2. Syntaxe

jQuery utilise le signe **\$** en guise d'appel. À chaque fois que vous verrez le signe **\$** dans du JavaScript, cela veut dire que c'est à jQuery qu'on fait appel. C'est là toute la subtilité de jQuery : ce n'est pas un nouveau langage en soi, bien qu'il possède sa propre syntaxe, mais il coexiste avec JavaScript !

Quelques exemples :

```
// Javascript ici :)
document.getElementById('maDiv').addEventListener('click', function() {
  alert('Hello');
});
// jQuery ici :)
$('#maDiv').click(function() {
  alert('Hello');
});
```

```
// Vérifier si un de mes éléments a bien une classe :
document.getElementById('maDiv').classList.contains('maClasse');
$('#maDiv').hasClass('maClasse');
```

```
// Afficher ou masquer un élément :
document.getElementById('maDiv').style.display = 'none';
$('#maDiv').hide();
document.getElementById('maDiv').style.display = 'block';
$('#maDiv').show();
```

```
// Gérer le hover sur un élément :
$('#maDiv').hover(function(){
    // Faire un truc au survol
}, function() {
    // Faire un truc quand on sort
});
```

3. Fonctions utiles

- **.click**(function() { /* Action à effectuer * / }); permet de gérer un **évènement** (le click)
- **.change**(function() { /* Action à effectuer * / }); permet de gérer un **évènement** (le change)
- **.html**(/* Contenu à écrire */); permet d'écrire dans l'élément cible. Si on ne lui passe pas d'argument, html() nous renvoie le contenu **déjà présent**.
- **.data**(/* nom de ma data */); permet d'accéder à l'attribut data que l'on a défini sur l'élément cible
- **.each**(function() { /* Action à effectuer * / }); permet d'initier une **boucle** sur un tableau. À chaque itération, on aura accès au sélecteur **\$(this)**, qui correspondra à l'itération en cours
- **.addClass('classeAAjouter') / .removeClass('classeAEnlever')**; permet d'ajouter ou d'enlever une classe à notre élément. Pratique pour rapidement ajouter ou enlever des propriétés CSS !