



jQuery (Débutant)

24 Mars 2010 @ SUPINFO Orléans

Qu'est-ce que jQuery ?

- jQuery est un framework JavaScript parmi tant d'autres (*Mootools, Prototype, YUI etc...*)
- Son objectif: Changer les méthodes avec lesquelles vous écrivez du javascript
- Simplifie l'écriture de code complexe écrit en js pur.

Ses avantages

- Cross browser (*IE 6.0+, FF 2.0+, Safari 2.0+, Opera 9.0+*)
- Support des sélecteurs CSS 3 (*Draft*)
- Léger ~24KB en version minified (*jquery.com*)
- Documentation complète et communautés actives (*plugins, exemples, etc ...*)

Comment commencer ?

```
<html>
<head>
<script type="text/javascript" src="path/to/jquery.js"></script>
<script type="text/javascript">
    $(document).ready(function() {
        alert("Welcome to JQuery");
    });
</script>
</head>
<body>
<a href="http://jquery.com/">jQuery</a>
</body>
</html>
```

- Simplement en ajoutant dans le header de votre code HTML l'import du fichier js
- Exécuter votre code dans la méthode Document.Ready

Utilisation du selecteur

- Sélectionner avec \$ ou jQuery -> *jQuery.noConflict()*;
- Sélectionner par ID -> `$("#monId")`
- Sélectionner par type -> `$("div")`
- Sélectionner par CSS -> `$(".maClass + div, p > span")`
- \$ ou jQuery retourne un tableau des éléments du DOM

Manipuler le DOM - Partie #1

- `$('#div.section').addClass('highlighted')`
- `$('#img.photo').attr('src', '/default.png');`
- `$('#a.foo').html('Click me now!');`
- `$('#p:odd').css('background-color', '#ccc');`

Manipuler le DOM - Partie #2

- `$('#div.section').next();`
- `$('#div.section').prev();`
- `$('#div.section').prev('a');`
- `$('#div.section').parent();`
- `$('#div.section').parents();`

Chaînage

- La plupart des méthodes jQuery retournent un objet jQuery qui représente souvent la même collection. Ce qui signifie que vous pouvez enchaîner les méthodes entre elles
- Exemple:
`$('div.section').hide().addClass('gone');`

Les événements

- Pas de javascript dans les balises HTML !
- Binder des actions à des événements simplement

- Exemple:

```
var message = 'Spoon!';  
$('#foo').bind('click', function() {  
    alert(message);  
});
```

Les performances

- Les performances dépendent de votre code et non du framework
- jQuery parse le DOM à chaque utilisation de `$/Find`
- Utilisation des ID (*plus rapide*)
 - `$("input")` -> lent
 - `$("#monItem input")` -> rapide

AJAX

- jQuery possède un support excellent pour AJAX
- Il implémente des méthodes génériques Cross-Browser pour une mise en place facile d'un fonctionnement AJAX
- Voici quelques méthodes à utiliser pour faire de l'AJAX:
 - \$.get(url, params, callback)
 - \$.post(url, params, callback)
 - \$.getJSON(url, params, callback)
 - \$.getScript(url, callback)

Extensible (*Plugins*)

- jQuery est extensible grâce aux plugins qui peuvent ajouter des nouvelles méthodes à l'objet jQuery
 - UI (*Contrôles utilisateurs plus 'user friendly'*)
 - Interface (*Carousel, Drag and Drop, Thickbox, jQuery UI*)
 - Formulaire (*Manipulation plus aisée, gestion des types, erreurs, etc ...*)

Aller plus loin

- www.jQuery.com (*Officiel, EN*)
- www.learningjQuery.com (*EN*)
- docs.jquery.com (*Documentation, EN*)
- plugins.jquery.com (*Plugins, EN*)