

jQuery, "write less, do more"

- *“jQuery is a fast and concise JavaScript Library that simplifies HTML document traversing, event handling, animating, and AJAX interactions for rapid web development. jQuery is designed to change the way that you write JavaScript.”*, <http://jquery.com>, 2011
 - *Lightweight footprint* : 31KB en prod
 - *Cross-browser* : IE6+, FF2+, Safari3+, Opera9+, Chrome
 - *CSS3 compliant* : sélecteurs CSS1-3

jQuery, "write less, do more"

- *“jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript.”*, <http://jquery.com>, 2014

Avantages

- Manipulation aisée du DOM
 - Parcours grâce aux sélecteurs et filtres
 - Modification (création, suppression, modification)
 - Gestion des événements (click, dblclick, hover, ...)
- Simplification du dialogue client/serveur
 - Différentes méthodes AJAX, cross-browser
 - Traitement de flux (XML, JSON, ...)
- Extensible, nombreux plugins, jQuery-UI
 - effets, animations et composants pour des IHM plus dynamiques et ergonomiques

jQuery simplement

- Les étapes de création d'une application web
 - construire la structure de l'interface XHTML
 - charger les feuilles de styles CSS nécessaires
 - charger les librairies et tous les scripts applicatifs
- Le script initial déclare le point d'entrée d'exécution de la logique applicative
 - `$(document).ready(mainFunction)`
- **Séparation fond, forme, logique**
 - ex. la gestion événementielle est dans les scripts !

Les sélecteurs

- Opération de sélection : $\$(\dots) \Leftrightarrow \text{jQuery}(\dots)$
- Trois sélecteurs de base :

Sélecteur	CSS*	jQuery	Sélectionne ...
Nom de balise	p	<code>\$('p')</code>	... tous les paragraphes du document
Identifiant	<code>#un-id</code>	<code>\$('#un-id')</code>	... l'élément <u>unique</u> du document dont l'id est un-id
Classe	<code>.une-classe</code>	<code>\$('.une-classe')</code>	... tous les éléments du document dont la classe est une-classe

(*) CSS Level 1 à 3 : <http://www.w3.org/Style/CSS>

L'objet jQuery

- La fonction `$('p')` crée un objet jQuery :
 - contient une collection d'éléments
 - offre toutes les méthodes jQuery dont l'action se fait **implicitement** sur toute la collection

- Modifier un ou plusieurs noeuds facilement

```
$( 'p' ).html( 'hello' );  
$( 'p' ).append( '<hr/>' );
```

- Gestion des attributs

```
$( 'p:first' ).addClass( 'color1' );  
$( 'a' ).attr( 'href', 'http://www.comem.ch' );
```

Encore des sélecteurs ...

- Tous les sélecteurs des specs CSS 1 à 3

<http://www.w3.org/TR/css3-selectors/#selectors>

`$('p:not(.myclass) ');` (pseudo-classe)

tous les § qui ne possède pas la classe `.myclass`

`$('a[href^=mailto:] ');` (sélecteur d'attribut)

tous les liens dont l'attribut href commence par mailto:

- Sélecteurs personnalisés de jQuery

<http://api.jquery.com/category/selectors/jquery-selector-extensions/>

`$('div.horizontal:eq(1) ')` (~ pseudo-classe)

sélection du 2^{ième} élément div possédant la classe `.horizontal`

`/!\` numérotation d'indice JS vs. numérotation CSS

Toujours des sélecteurs ...

- Sélecteurs pour formulaire

- `:text` `:checkbox` `:radio` `:input` `:button` `:enabled`
`:disabled` `:checked` `:selected`

- `$(':radio:checked')` sélection des radios cochés

- `$(':radio, :checkbox')` sélection des radios et des checkbox

- Méthodes de parcours du DOM

- <http://api.jquery.com/category/traversing/tree-traversal/>

- `.next()` : sélectionne l'élément frère suivant

- `.nextAll()` `.prev()` `.prevAll()` `.parent()` `.children()` ...

- `.filter(selector)` `.filter(function)`

Modification du DOM, en bref ...

- Créer des éléments HTML : `$()` ex. `$('<div/>');`
- Insérer dans : `.append()` `prepend()` ...
- Insérer à côté : `.insertAfter()` `.insertBefore()` ...
- Insérer autour : `.wrap()`, `.wrapAll()` ...
- Remplacer : `.html()`, `.text()`, `.replaceAll()` ...
- Supprimer dans : `.empty()`
- Supprimer : `.remove()`

<http://api.jquery.com/category/manipulation/>

Gestion des événements

- Ajout facile d'événements simples

```
$( 'a' ).click(function(evt) {  
    alert("Go to " + this.href);  
});
```

```
$( 'p' ).dblclick(function(evt){  
    $(this).hide();    // cacher le §  
});
```

- contexte d'exécution du comportement : `this`
- et l'habituel objet événement standardisé (`evt`)

Des événements ...

- Événements simples :

- <http://api.jquery.com/category/events/>

<code>blur</code>	<code>error</code>	<code>keyup</code>	<code>mouseout</code>	<code>scroll</code>
<code>change</code>	<code>focus</code>	<code>load</code>	<code>mouseover</code>	<code>select</code>
<code>click</code>	<code>keydown</code>	<code>mousedown</code>	<code>mouseup</code>	<code>submit</code>
<code>dblclick</code>	<code>keypress</code>	<code>mousemove</code>	<code>resize</code>	<code>unload</code>

- Événements composés :

- `toggle(function, function, [function ...])`

- cycle de comportement à chaque clic

- `hover(function, function)`

- gestion du survol souris d'un élément (over/out)