

Programmation Web Javascript et JQuery

Julien David

A101 - david@lipn.univ-paris13.fr

2016

Langage Javascript :

- Langage de script, interprété par le navigateur.
- permet de rendre les pages webs plus dynamiques.
- permet de modifier le code HTML et le code CSS d'une page.
- permet un gestion des événements plus fine que CSS.

JQuery :

- bibliothèque javascript permettant de simplifier la syntaxe.

Citation (d'un illustre inconnu) :

“**Javascript** is to **Java** as **Hamster** is to **Ham**.”

(le javascript est au java ce que le hamster est au jambon).

Code Javascript

- Compris entre les balises `<script>` et `</script>` à l'intérieur du HTML.
- Les instructions terminent par ;
- Commentaires :
 - // Commentaire uniligne
 - /* Commentaire multiligne */

Trois méthodes :

- 1 écrire le code directement dans le fichier HTML.
- 2 écrire le code dans un fichier à part et inclure le code.
Dans les deux cas, on utilise la balise script
- 3 associer une balise à un événement.

Jquery

Pour inclure la bibliothèque JQuery, on ajoute dans l'en-tête du HTML la commande : `<script src="jquery-2.2.1.min.js"> </script>`

Plusieurs instructions

- alert
- write
- innerHTML (on ne l'utilisera pas)
- log

Les variables

- Déclaration : on ajoute *var* devant le nom de la variable.
- le type d'une variable :
 - est implicite (pas besoin de déclarer une variable).
 - dépend de la valeur qu'on lui affecte (peut varier dans le temps).
- si une variable n'est pas initialisée, elle est égale à *undefined*

Les types de variables

- nombres
- booléens,
- chaînes de caractères,
- objet,
- fonction.

Pour bien coder, on garde les règles habituelles

- toujours déclarer les variables, avec *var*, au début du code.
- toujours initialiser les variables.

La concaténation

Il est possible de concaténer des chaînes de caractères entre elles, ou avec des variables.

Opérateur : +

Attention : symboles et conversion automatique

- Le symbole + signifie donc addition pour les nombres et concaténation pour les chaînes de caractères.
- Les symboles -, * et / signifient soustraction, multiplication et division pour les nombres ET pour les chaînes de caractères.

Les tableaux en Javascript

- un tableau associe un entier ...
- ... à une valeur de n'importe quel type.

Boucles for et while

Le principe et la syntaxe sont les même qu'en *C*.

if, else if, else, switch

Le principe est le même qu'en *C/C++*.

La différence (qu'on retrouve en PHP)

- Opérateur supplémentaire : `===` teste si la valeur et le type sont égaux.

On préférera utiliser cet opérateur plutôt que `==`.

Syntaxe

```
function nomFonction(paramètres){ }
```

Particularités

- on peut définir une fonction sans lui donner de nom,
- on peut définir une fonction sans déclarer le nombre exact de paramètres. On utilise alors l'objet *arguments* pour récupérer les arguments de la fonction.

“Objets”

- En javascript, les objets sont des variables contenant plusieurs valeurs.
- On ne peut pas définir un objet et réutiliser cette définition.
- On peut déclarer une variable de type objet et initialiser ses champs.
- On peut à tout moment ajouter un champ dans un objet.
- On peut définir une fonction qui joue le rôle de constructeur (qui joue alors le rôle d'une définition).

Les objets prédéfinis

- Window,
- Screen,
- Location,
- History,
- Navigator,
- Timing,
- Cookies.

Des objets dépréciés

Ne pas utiliser *Number*, *String*, *Boolean* et *Object*.

Accéder à une balise HTML

- Javascript permet de modifier dynamiquement des zones d'une page HTML,
- Pour cela, on veut pouvoir accéder à des zones spécifiques de la page afin de pouvoir les modifier.

```
<p id="idPar" class="classPar">Je suis un paragraphe.</p>
```

Sélecteurs JQuery

La syntaxe est la même qu'en css pour désigner les balises, les classes et les identifiants.

<code>\$("p")</code>	Accède à toutes les balises de type <i>p</i>
<code>\$(".classPar")</code>	Accède à toutes les balises de classe <i>classPar</i>
<code>\$("#idPar")</code>	Accède à la balise identifiée par <i>idPar</i>

jQuery : les sélecteurs

```
<p id="idPar" class="classPar">Je suis un paragraphe.</p>
```

Accéder à des balises en Javascript

```
document.getElementsByTagName("p")  
document.getElementsByClassName("classPar")  
document.getElementById("idPar")
```

Accéder aux même balises en JQuery

```
$("p")  
$(".classPar")  
$("#idPar")
```

Fonctions de JQuery

Certaines fonctions en JQuery peuvent :

- Permettre d'accéder à une valeur (la fonction ne prend pas de paramètres)
- De modifier une valeur (la fonction prend un ou plusieurs paramètres)

Fonctions de JQuery

text()	Lire/Initialiser le contenu texte d'une balise
html()	Lire/Initialiser le code HTML d'une balise
css()	Lire/Initialiser le code CSS associé à une balise
attr()	Lire/Initialiser la valeur de l'attribut d'une balise

Fonctions de JQuery : ajout

<code>append()</code>	Ajoute du contenu après le contenu d'une balise
<code>prepend()</code>	Ajoute du contenu avant le contenu d'une balise
<code>after()</code>	Ajoute du contenu après une balise
<code>before()</code>	Ajoute du contenu avant une balise
<code>addClass()</code>	Ajoute une class à une balise

Fonctions de JQuery : suppression

<code>remove()</code>	Supprime une balise et son contenu
<code>empty()</code>	Supprime le contenu d'une balise
<code>removeClass()</code>	Supprime une classe d'une balise

- `remove()`
- `removeClass()`
- `empty()`

Les événements

Pour chaque balise du code HTML, il est possible :

- de lui associer une fonction à exécuter (*callback*) pour un événement donné,
- mais aussi d'associer un callback à exécuter une fois l'évènement terminé.
- Enfin, il est possible de supprimer un callback associé à un événement (méthode *off*).

JQuery : les événements

Souris	Clavier	Formulaire	Document/Fenêtre
click dblclick mouseenter mouseleave hover	keypress keydown keyup	submit change focus blur	ready resize scroll

Cette liste n'est pas exhaustive.

Fonctions de JQuery : déplacement dans le DOM (non exhaustif)

parent()	accède à la balise parent de celle sélectionnée
children()	accède aux balises incluses dans celle sélectionnée
find()	accès ciblé aux balises incluses dans celle sélectionnée
siblings()	accède aux balises ayant le même parent que celle sélectionnée
next()	accède aux <i>sibling</i> suivant
prev()	accède aux <i>sibling</i> précédent

Fonctions de JQuery : les effets

- show()
- hide()
- toggle()
- fadeIn()
- fadeOut()
- fadeToggle()
- slideUp()
- slideDown()
- slideToggle()