

Développement Web

JavaScript

Halim Djerroud <hdd@ai.univ-paris8.fr>

LIASD - Université Paris 8

2018

Plan de cours général

8 semaines de cours :

- 1 Introduction au web dynamique
- 2 Gestion de rendu et décoration
- 3 Le duo magique : PHP et MySql
- 4 Programmation OO et les design patterns
- 5 **TP noté pour faire une pause et un peu de JS avec ses saveurs (jQuery).**
- 6 Les frameworks et Symfony4
- 7 Symfony4 et ses rouages
- 8 Correction de projets

Plan de cours

- 1h30 Cours
- 2h00 Travaux pratiques.
 - 1 JavaScript
 - 2 JQuery

Support de cours

- <http://djerroud.halim.info/index.php/cours/programmation-web/>

Introduction JavaScript

- Syntaxe proche du langage C
- Comme PHP, il offre des types de base mais fait faiblement typé
- Implémente un modèle objet rudimentaire (et donc quelques objets de base)
- Pas d'entrées/sorties

Intérêt de JavaScript

Code

```
<html>
  <head>
    <title>Page statique</title>
  </head>
  <body>
    <div>
      Nous sommes le 2/10/2008
    </div>
  </body>
</html>
```

Intérêt de JavaScript

Code

```
<html>
  <head>
    <title>Page dynamique</title>
  </head>
  <body>
    <script type = "text/javascript">
      date = new Date();
      document.writeln("Nous sommes le ", date);
    </script>
  </body>
</html>
```

Script

- Portion de code qui vient s'insérer dans une page HTML
- Le code du script n'est toutefois pas visible dans la fenêtre du navigateur car il est compris entre des balises spécifiques qui signalent au navigateur qu'il s'agit d'un script écrit en langage JavaScript
- Interprété du côté client

Intégration de JavaScript

- Dans la balise `<SCRIPT>`
- Dans un fichier externe `<SCRIPT SRC=" fichier .js">`
- Dans un attribut de balise
- En gestion d'événements

L'interprétation du code JavaScript se fait au fur et à mesure de sa lecture dans la page ou lors du déclenchement d'événements par l'utilisateur.

Exemple

Code

```
<html>
  <head>
    <title>Glups</title>
  </head>
  <body>
    <h1>Glups</h1>
    Avec un bon navigateur
    <script language="JavaScript">
      document.writeln("(comme "+navigator.appName+")")
    </script>
    on peut réaliser des documents
    personnalisés et dynamiques.
  </body>
</html>
```

Les Types de variables

Type de variables :

- Entiers, long ex : $a = -100$; $b = 10$;
- Flottants, double ex : $a = 0.123$; $b = -0.4e5$; $c = .67E - 89$
- Strings ex : $t1 = \text{"texte"}$; $t2 = \text{'texte'}$
- Boolean ex : $b = \text{false}$;

Typage et Variable

Pas de déclaration des variables

```
nbr = 10;  
fl = 3.141;  
str1 = "L'étoile";  
str2 = 'brille';  
lien = '<a href="index.htm">Home</a>';
```

Pas de déclaration des variables

```
locale (uniquement dans le script ou la fonction)  
=> var vloc = 0 ;  
globale (en tout point du document)  
=> vglob = 0 ;
```

Les expressions

- arithmétique $(3+4) * (56.7 / 89)$
- chaîne de caractères "L'étoile" + " " + "filante"
- logique $temp == 37$ h2o = (temp<100)? "eau" : "vapeur";
h2o = (temp>0)? ((temp<100)? "eau" : "vapeur") : "glace"

Les opérateurs

- affectation
+=, -=, *=, /=, %=, &=, |=, <<=, >>=
- comparaison
==, !=, <, <=, >, >=
- arithmétique
%, ++, -
- logique (évaluation paresseuse)
&&, ||, !
- bit
&, |, ^ (XOR), <<, >>, >>>

Structures de contrôle

Type de variables :

- if else
- switch case
- for
- while
- break
- continue
- do while

Fonctions

Définition

```
function nomfonction(param1, ..., paramN) {  
  // code JavaScript  
  return expression ;  
}
```

Appel

```
nomVariable = nomfonction(exp1, ..., expN);  
passage des paramètres par valeur
```


Arguments

Code

```
function somme() {  
    var argv = somme.arguments;  
    var argc = somme.arguments.length;  
    var result = 0;  
    for (var i = 0 ; i < argc ; i++) {  
        result += argv[i];  
    }  
    return result;  
}  
  
somme(1,2,3) retourne 6 et somme(2) retourne 2
```

Lire / Écrire

- `prompt()`
Ouvre une boîte de dialogue avec une zone saisie et 2 boutons OK et Annuler, retourne l'information lue
- `confirm()`
Ouvre une boîte de dialogue avec 2 boutons OK et Annuler, retourne un booléen
- `alert()`
Permet d'écrire un message dans une fenêtre

Les évènements

- `onclick` : un clic du bouton gauche de la souris sur une cible
- `onmouseover` : passage du pointeur de la souris sur une cible
- `onblur` : une perte de focus d'une cible
- `onfocus` : une activation d'une cible
- `onselect` : une sélection d'une cible
- `onchange` : une modification du contenu d'une cible
- `onsubmit` : une soumission d'un formulaire
- `onload` : un chargement d'une page
- `onunload` : la fermeture d'une fenêtre ou le chargement d'une page autre que la courante

Les événements

Code

```
<a href="" onclick="alert('Bonjour')">Cliquez</a>
```