



Sébastien Lafontaine
Version 1.1.0
22/01/2018
Javascript

Table des matières

LES CONDITIONS	2
utiliser les if, else les switch	
LES BOUCLES	3
répétition des variables	
LES TABLEAUX	3
ajout, suppression, modification et création	
LES FONCTIONS	4
création des fonctions	
LES OBJETS	5
LIEN ENTRE HTML ET JS	5

Javascript = code client side

Concaténation = ajout de valeur d'une **variable** l'une à l'autre

```
(ex:  var mot1 = chat ;  
      var mot2 = chien ;  
      var mot3 = mot1 + mot2 ;  
)
```

|| = ou

&& = et

!qqc = l'inverse

```
(ex:  var un = 1 ;  
      var deux = 2 ;  
      var trois = un + deux ;  
      var condition = un > deux ; → false  
      var resultat = !condition ; → true  
)
```

Les conditions

possibilité d'exécution du code différemment selon ce que l'on veut

```
var uneVariable = 'Bonjour' ;
```

```
if('uneVariable') {  
    alert('ceci est plein') ;  
} else {  
    alert('ceci est vide') ;  
}
```

Cette structure conditionnelle est correcte mais va poser problème lorsque l'on ajoutera des 'else if'.

Switch : la structure conditionnelle qui ne fait pas des rallonges

```
var mot4 = message ;
```

```
switch(uneVariable) {  
    case 'chat' :  
        message = 'joli les petits chats' ;  
        break ;  
  
    default :  
        message = 'je ne connais pas cet animal' ;  
}
```

```
alert(message);
```

```
if(uneVariable == 'chat') {
  alert('comme moi');
} else {
  alert('pas comme moi');
}
```

→

```
alert((uneVariable == 'chat') ?
'comme moi' : 'pas comme moi'
);
```

Les boucles

répétition de code et de condition sur une durée déterminée

++ = incrément de +1

-- = incrément de -1

```
Var nb = 1 ;
```

```
while (nb < 30) {
  Nb ++;
}
```

NaN = not a number

```
for (var repete = 0 ; repete < 10 ; repete++) {
  alert('coucou');
}
```

→ on remplace très souvent cette valeur de répétition par 'i'.

```
for (var i = 0 ; i < 10 ; i++) {
  alert('coucou');
}
```

Les tableaux

insérer des données dans des tableaux afin de les utiliser par la suite

```
var unTableau = [ 'chien', 'chat', 'lapin', 'canard' ];
                 0      1      2      3
```

```
unTableau[2] = 'vache';
```

→ la troisième case du tableau va être remplacée ('lapin' par 'vache')

Ajouter des valeurs:

`unTableau.push('cochon', 'renard');` → ajout de 'cochon' et 'renard' au tableau après 'canard'

`unTableau.pop();` → enlève la dernière valeur du tableau, donc renard

(ex :
`var animaux = [];`
`while (true) {`
 `var promptText = prompt('entrer un animal');`
 `if(promptText != '') {`
 `animaux.push(promptText);`
 `} else {`
 `break;`
 `}`
`}`

`alert(unTableau.indexOf('chat'));`
→ Afficher une valeur du tableau par son index (1)

Supprimer des valeurs et remplacer :

`unTableau.splice(placement, nbéléments, 'élémentajouté', 'élémentajouté');`
On peut initialiser le nombre d'éléments à 0 pour ne rien supprimer.

ex : `unTableau.splice(1, 2, 'mouton', 'poule');`

`['chien', 'chat', 'mouton', 'poule']` On remplace 'lapin' et 'canard' du premier tableau par 'mouton' et 'poule' après 'chat'.

Les fonctions

suite d'opération retournant des résultats configurés.

Affichage :

```
function bonjour(){
    alert('bonjour tout le monde');
}
```

`bonjour();`

Addition :

```
var result(number(nb1) + number(nb2));
```

```
Function addition(nb1, nb2){
    return result
}
```

```
var prompt1 = prompt('un premier nombre : ');
var prompt2 = prompt('n deuxième nombre : ');
```

```
addtion(prompt1, prompt2);
```

Les objets

ensemble des données utiles du code.

modèle n°1 :

```
var utilisateur = {nom : 'Jean', age : '22', sexe : 'M'};
console.log(utilisateur.nom);
```

```
var utilisateur = {nom : {prenom : 'Jean', famille : 'Stalin'}, age : '22', sexe : 'M'};
console.log(utilisateur.nom);
→ objet imbriqué
```

JSON : javascript object notation

modèle n°2 : (plus utilisé)

```
function Personne (nom, age, sexe, cm) {
    this.nom = nom ;
    this.age = age ;
    this.sexe = sexe ;
    this.taille = cm ;
}
```

On appelle la fonction et on lui place des paramètres

déclaration des objets

```
var utilisateur1 = new Personne('Jean', '22', 'M');
var utilisateur2 = new Personne('Alain', '32', 'M');
```

Création des nouveaux users

```
utilisateur2.nom = 'Michel';
```

Si on veut changer une info

```
console.log(utilisateur1);
console.log(utilisateur2);
```

Récupération de données HTML

faire le lien entre le js et les données de la page html

HTML

```
<h1 id='titre'>Regardez un titre</h1>
```

JS

```
var titre = document.getElementById('titre.innerHTML');
```

on récupère l'id de la page html

HTML

```
<p id='parag'>Oh un paragraphe</p>  
<p id='parag2'>Oh un paragraphe</p>
```

JS

```
var paragraphes = document.getElementsByTagName('p');
```

```
for(var i = 0 ; i < paragraphes.length ; i++){  
    console.log(paragraphes[i].innerHTML);  
}
```