

HTML et JavaScript

Exploration du *Document Object Model* (DOM)

"The W3C Document Object Model (DOM) is a platform and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure, and style of a document."

Le Javascript peut manipuler tous les éléments de la page HTML : balise, attribut, évènement.
L'accès est de type *objet* : Les éléments ont des méthodes (des actions) et des propriétés (des valeurs).

1) Comment changer le contenu d'un élément (d'une balise HTML) :

On considère l'élément suivant : `<div id="aaa" class="bbb"> texte affiché </div>`
On note 2 attributs : **id** (nom « aaa » unique dans la page), et **class** (lié à une css, plusieurs balises peuvent l'utiliser).

a) (DOM2) La méthode la plus simple est l'accès à l'**id** unique directement en tant qu'objet :

```
aaa.innerHTML="nouveau texte à afficher";
```

b) (DOM2/ECMA-Sc6) La méthode la plus fiable est l'accès à l'**id** unique, en utilisant la méthode *querySelector*:

```
document.querySelector("#aaa").innerHTML="nouveau texte à afficher";

ou :
var x=document.querySelector("#aaa") ; // x est un objet représentant l'élément dont l'id=aaa
x.innerHTML="Nouveau texte à afficher" ; // on utilise la propriété innerHTML de l'objet
```

c) Autres méthodes : celles qui renvoient une liste d'éléments (nb : ...*Elements*...)

Accès par le nom de la balise (TagName) :

```
var x=document.querySelectorAll("div") ; // DOM2/ECMA-Sc6
ou
var x=document.getElementsByTagName("div") ; // x est un tableau d'éléments
x[0].innerHTML="Nouveau texte à afficher" ; // on utilise le premier élément du tableau (1er <div>)
```

Accès par le nom de l'attribut *class* :

```
var x=document.querySelectorAll(".bbb") ; // DOM2/ECMA-Sc6
ou
var x=document.getElementsByClassName("bbb") ; // x est un tableau d'éléments
x[0].innerHTML="Nouveau texte à afficher" ; // on utilise le premier élément du tableau
```

Autre exemple de propriété : `x.length` qui donne le nombre d'éléments trouvés.

Remarque : **innerHTML** est une propriété de balise d'affichage texte.

Pour une balise `<input>` ou `<button>`, les propriétés sont différentes suivant l'endroit où le texte doit être affiché. Exemple :

```
x.value = "Envoyer" ; //pour modifier ce qui est écrit sur le bouton <input type='text'>
x.innerHTML = "Envoyer" //pour modifier ce qui est écrit sur le <button>
```

2) Comment changer le style d'un élément (d'une balise HTML) :

```
document.querySelector("#aaa").style.color= "blue" ;
ou
aaa.style.color= "blue" ;
```

On utilise l'attribut *style* (qui est lui-même un objet représentant le css) et sa propriété *color*.

3) Autres fonctions liées aux éléments HTML :

Les sites de références donnent les méthodes pour gérer (afficher, modifier, comparer) chaque élément d'une page HTML. Chaque balise est considérée comme un nœud (NODE) hiérarchisé, donc accessible.

Ex : Modifier une page HTML en ajoutant une balise <p> à une balise <div id='div1'> :

```
<div id="div1">
< p id="p1">This is a paragraph.</p>
< p id="p2">This is another paragraph.</p>
< /div>
```

← La nouvelle balise sera ajoutée ici

```
<script>
var para = document.createElement("p"); // Créé une balise <p>
var texte = document.createTextNode("Je suis la nouvelle balise");// Créé une zone texte
para.appendChild(texte); // Ajoute le texte à la balise <p> créée

var cible = document.querySelector("#div1");
cible.appendChild(para); // Ajoute la balise <p> à l'endroit voulu dans la page html
</script>
```

4) Autres objets du DOM :

a) Navigator :

Cet objet représente le navigateur. Exemple :

```
var x = "Browser Name: " + navigator.appName;
```

b) Window :

Cet objet représente une fenêtre ouverte dans le navigateur.

On y retrouve les propriétés et les méthodes utilisée directement en JavaScript, comme *alert()*, *confirm()*, *prompt()*.

Normalement, l'appel de ces méthodes devraient être : `window.alert(" Message !!!")`

Quand on n'indique pas la *window*, on considère qu'il s'agit de la fenêtre courante.

Exemples d'autres méthodes de l'objet **window** :

```
function ouvreFenetre()
{
myWindow = window.open('', '', 'width=200,height=100'); // Ouvre une nouvelle fenetre
myWindow.document.write("<p>This is 'myWindow'</p>"); // Du texte quelconque
}

function deplaceFenetre()
{
myWindow.moveTo(0,0); // Deplace la nouvelle fenetre
myWindow.focus(); // La souris est dans la nouvelle fenetre
}
```

Sites de référence recommandés :

<http://www.w3schools.com>
<http://www.startyourdev.com/>