

Le DOM

Le Modèle Objet du Document

Traduction d'un document anglais de Christian Heilmann :

« The essentials of DOM scripting in 10 minutes »,
<http://icant.co.uk/articles/domessentials/>

Le DOM – Décortiquer le Document

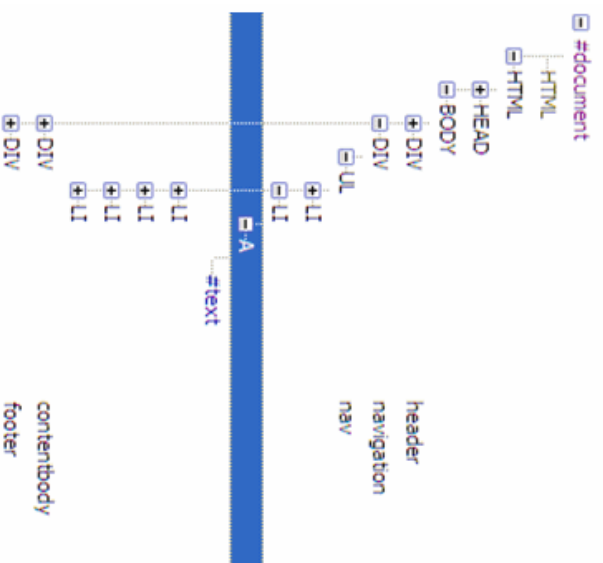
- Le Modèle Objet du Document est une représentation du document tout entier sous forme de nœuds et d'attributs
- On peut accéder à chacun de ces nœuds et attributs et les changer ou les supprimer
- On peut aussi créer de nouveaux nœuds ou ajouter des attributs à ceux qui existent
- En d'autres mots, **le document est à vous !**

Un exemple HTML

```
<div id="header">
  
</div>
<div id="navigation">
  <ul id="nav">
    <li><strong>Home</strong></li>
    <li><a href="articles.html">Articles</a></li>
    <li><a href="about.html">About Me</a></li>
    <li><a href="photos.html">Photos</a></li>
    <li><a href="clients.html">Clients</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</div>
```

L'arbre du DOM

- Le navigateur le voit comme une liste de nœuds imbriqués, d'attributs et de nœuds enfants
- Un nœud peut être un élément ou un texte ou un commentaire ou bien d'autres choses (un attribut, ...)



Les méthodes de récupération du DOM

- `document.getElementById(ID)`
retourne un élément sur base de son ID
- `document.getElementsByTagName(name)`
retourne un tableau de tous les éléments qui ont le nom de balise spécifié

Récupérer un élément sur base de son ID

```
<div id="header">  
  x = document.getElementById('nav');  
    
</div>  
  
<div id="navigation">  
  <ul id="nav">  
    <li><strong>Home</strong></li>  
    <li><a href="articles.html">Articles</a></li>  
    <li><a href="about.html">About Me</a></li>  
    <li><a href="photos.html">Photos</a></li>  
    <li><a href="clients.html">Clients</a></li>  
    <li><a href="contact.html">Contact</a></li>  
  </ul>  
</div>
```

Récupérer des éléments sur base de leur nom de balise (TagName)

```
<div id="header">  
    
</div>  
<div id="navigation">  
  <ul id="nav">  
    <li><strong>Home</strong></li>  
    <li><a href="articles.html">Articles</a></li>  
    <li><a href="about.html">About Me</a></li>  
    <li><a href="photos.html">Photos</a></li>  
    <li><a href="clients.html">Clients</a></li>  
    <li><a href="contact.html">Contact</a></li>  
  </ul>  
</div>
```

Récupérer des éléments sur base de leur nom de balise (TagName)

- Comme le résultat de
getElementsByTagName (name) est un tableau,
on peut atteindre chacun des éléments
grâce à son indice

Ex. getElementsByTagName ("h1") [0]
cible le premier titre h1 du document

- On peut aussi savoir combien d'éléments avec un
nom de balise donné il y a dans le document
en lisant la propriété « length » du tableau

Ex. getElementsByTagName ("h1").length
donne le nombre de titres h1 du document

Récupérer des éléments sur base de leur nom de balise (TagName)

```
x = document.getElementsByTagName ( 'li' );  
o = x[3] ;  
  
<div id="navigation">  
  <ul id="nav">  
    <li><strong>Home</strong></li>  
    <li><a href="articles.html">Articles</a></li>  
    <li><a href="about.html">About Me</a></li>  
    <li><a href="photos.html">Photos</a></li>  
    <li><a href="clients.html">Clients</a></li>  
    <li><a href="contact.html">Contact</a></li>  
  </ul>  
</div>
```

Frères précédent et suivant

- Une fois qu'on a atteint un élément, on peut utiliser cet élément comme point de départ
- On peut alors récupérer les éléments qui sont au même niveau dans l'arbre logique avec `nextSibling` (« frère suivant ») et `previousSibling` (« frère précédent »)
- On peut atteindre l'élément qui contient l'élément courant avec `parentNode` (« nœud parent »)

Frères précédent et suivant

```
x = document.getElementById('li');
o = x[3];
p = o.previousSibling;

</div>

<div id="navigation">
  <ul id="nav">
    <li><strong>Home</strong></li>
    <li><a href="articles.html">Articles</a></li>
    <li><a href="about.html">About Me</a></li>
    <li><a href="photos.html">Photos</a></li>
    <li><a href="clients.html">Clients</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</div>
```

Frères précédent et suivant

```
x = document.getElementById('li');
o = x[3];
p = o.nextSibling;

</div>

<div id="navigation">
  <ul id="nav">
    <li><strong>Home</strong></li>
    <li><a href="articles.html">Articles</a></li>
    <li><a href="about.html">About Me</a></li>
    <li><a href="photos.html">Photos</a></li>
    <li><a href="clients.html">Clients</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</div>
```

Frères précédent et suivant

```
<
  x = document.getElementById('li');
  o = x[3];
  p = o.parentNode;
<
```

```
<div id="navigation">
```

```
<ul id="nav">
```

```
<li><strong>Home</strong></li>
```

```
<li><a href="articles.html">Articles</a></li>
```

```
<li><a href="about.html">About Me</a></li>
```

```
<li><a href="photos.html">Photos</a></li>
```

```
<li><a href="clients.html">Clients</a></li>
```

```
<li><a href="contact.html">Contact</a></li>
```

```
</ul>
```

```
</div>
```

Pensez aux Enfants !

- Si un élément contient d'autres éléments, alors ces éléments sont ses enfants
- `childNodes` est une liste (un tableau) de tous les nœuds enfants directs de l'élément
- N.B. Il ne descend pas aux sous-niveaux, il ne reprend donc pas les descendants
- On peut cibler un élément enfant de l'élément courant via l'indice ou via la méthode `item()`

Pensez aux Enfants !

- Les propriétés raccourcies
monElement.firstChild et
monElement.lastChild
sont en fait des versions simplifiées de
monElement.childNodes[0] et
monElement.childNodes[
monElement.childNodes.length-1]
- On peut vérifier si un élément a des enfants
en appelant la méthode hasChildNodes()
qui retourne un Booléen (true ou false)

Accéder aux Enfants

```
<div id="navigation">  
  <ul id="nav">  
    <li><strong>Home</strong></li>  
    <li><a href="articles.html">Articles</a></li>  
    <li><a href="about.html">About Me</a></li>  
    <li><a href="photos.html">Photos</a></li>  
    <li><a href="clients.html">Clients</a></li>  
    <li><a href="contact.html">Contact</a></li>  
  </ul>  
</div>
```


Accéder aux Enfants

```
<div>  
  <div id="navigation">  
    <ul id="nav">  
      <li><strong>Home</strong></li>  
      <li><a href="articles.html">Articles</a></li>  
      <li><a href="about.html">About Me</a></li>  
      <li><a href="photos.html">Photos</a></li>  
      <li><a href="clients.html">Clients</a></li>  
      <li><a href="contact.html">Contact</a></li>  
    </ul>  
  </div>  
</div>
```

Accéder aux Enfants

```
<div>  
  <div id="navigation">  
    <ul id="nav">  
      <li><strong>Home</strong></li>  
      <li><a href="articles.html">Articles</a></li>  
      <li><a href="about.html">About Me</a></li>  
      <li><a href="photos.html">Photos</a></li>  
      <li><a href="clients.html">Clients</a></li>  
      <li><a href="contact.html">Contact</a></li>  
    </ul>  
  </div>  
</div>
```

Accéder aux Enfants

```
<div>  
  x = document.getElementById('nav');  
  o = x.childNodes;  
  p = o[3];  
  // or o.item(3);  
</div>
```

```
<ul id="nav">  
  <li><strong>Home</strong></li>  
  <li><a href="articles.html">Articles</a></li>  
  <li><a href="about.html">About Me</a></li>  
  <li><a href="photos.html">Photos</a></li>  
  <li><a href="clients.html">Clients</a></li>  
  <li><a href="contact.html">Contact</a></li>  
</ul>  
</div>
```

Châînage des méthodes du DOM

- On peut enchaîner les méthodes pour cibler plus rapidement des parties du document
- Exemples :
 - document.getElementById("nav").getElementsByTagName("li")[3].firstChild
retourne ... ?
 - document.
getElementsByTagName("li")[2].childNodes[3].
firstChild
retourne ... ?

Châînage des méthodes du DOM

```
<div id="navigation">
  <ul id="nav">
    <li><strong>Home</strong></li>
    <li><a href="articles.html">Articles</a></li>
    <li><a href="about.html">About Me</a></li>
    <li><a href="photos.html">Photos</a></li>
    <li><a href="clients.html">Clients</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</div>
```

Les propriétés des noeuds

- Une fois que vous avez atteint le nœud que vous souhaitez cibler, vous pouvez récupérer ses propriétés :
 - nodeType contient le type du nœud
 - 1 pour un nœud Element,
 - 3 pour un nœud Texte, ...
 - nodeName contient le nom du nœud
 - soit le nom de l'élément,
 - soit #textNode
 - nodeValue contient la valeur du nœud
 - null pour des nœuds Element et
 - le texte pour des nœuds Texte

Les propriétés des noeuds

```
<div id="nav">  
  x=document.getElementById('nav').getElementsByName('li')[3].firstChild;
```

```
</div>
```

```
<div id="navigation">
```

```
  <ul id="nav">
```

```
    <li><strong>Home</strong></li>
```

```
    <li><a href="articles.html">Articles</a></li>
```

```
    <li><a href="about.html">About Me</a></li>
```

```
    <li><a href="photos.html">Photos</a></li>
```

```
    <li><a href="clients.html">Clients</a></li>
```

```
    <li><a href="contact.html">Contact</a></li>
```

```
  </ul>
```

```
</div>
```

```
nodeName :A  
nodeType :1  
nodeValue :null
```

Les propriétés des noeuds

```
<div id="nav">  
  x=document.getElementById('nav').getElementsByTagName('a')[2].firstChild;
```

```
</div>
```

```
<div id="navigation">
```

```
  <ul id="nav">
```

```
    <li><strong>Home</strong></li>
```

```
    <li><a href="articles.html">Articles</a></li>
```

```
    <li><a href="about.html">About Me</a></li>
```

```
    <li><a href="photos.html">Photos</a></li>
```

```
    <li><a href="clients.html">Clients</a></li>
```

```
    <li><a href="contact.html">Contact</a></li>
```

```
  </ul>
```

```
</div>
```

```
nodeName :#text  
nodeType :3  
nodeValue :Photos
```

Changer la valeur des nœuds Texte

- On peut lire la nodeValue des nœuds Texte, mais on peut aussi la changer
- Quand on change la nodeValue, le texte du nœud change (le texte dans le HTML et l'affichage changent)
- Il est bon de s'assurer que le nœud que vous essayez de changer est bien un nœud Texte en vérifiant que son nodeType est bien 3

Changer la valeur des nœuds Texte

```
<div id="navigation">
  <ul id="nav">
    <li><strong>Home</strong></li>
    <li><a href="articles.html">Articles</a></li>
    <li><a href="about.html">>About Me</a></li>
    <li><a href="photos.html">Visuals</a></li>
    <li><a href="clients.html">Clients</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</div>
```

```
x=document.getElementById('nav').getElementsByTagName('a')[2].firstChild;
x.nodeValue = 'Visuals';
```

Lire et changer les attributs

- Les nœuds `Element` ont des attributs, que vous pouvez changer aussi
- `node.getAttribute(attribut)` retourne la valeur courante de l'attribut
- `node.setAttribute(attribut, valeur)` met la valeur de l'attribut à valeur
- Il n'y a pas besoin de créer l'attribut avant de lui attribuer une valeur

Changer les attributs

```
<div id="navigation">
  <ul id="nav">
    <li><strong>Home</strong></li>
    <li><a href="articles.html">Articles</a></li>
    <li><a href="about.html">About Me</a></li>
    <li><a href="photos.php">Photos</a></li>
    <li><a href="clients.html">Clients</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</div>
```

```
x=document.getElementById('nav').getElementsByTagName('li')[3].firstChild;
x.setAttribute('href', 'photos.php');
```

Créer un nouvel attribut

```
<div id="navigation">
  <ul id="nav">
    <li><strong>Home</strong></li>
    <li><a href="articles.html">Articles</a></li>
    <li><a href="about.html">About Me</a></li>
    <li><a href="photos.php" target="pop">Photos</a></li>
    <li><a href="clients.html">Clients</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</div>
```

Créer de nouveaux noeuds

- On peut créer de nouveaux noeuds avec 3 méthodes :
 - nouveauNoeud = node.cloneNode(bool)
créé nouveauNoeud en tant que copie (clone) de node
Si bool vaut true, le clone inclut des clones de tous les noeuds enfants de l'original
 - document.createElement(element)
créé un nouveau noeud Element
 - document.createTextNode(texte)
créé un nouveau noeud Texte

Ajouter de nouveaux nœuds au Document

- Les deux méthodes précédentes de création de nœud créent effectivement de nouveaux nœuds, et vous pouvez y appliquer d'autres méthodes du DOM, mais ces nœuds n'apparaissent pas encore dans le document
- Pour les faire apparaître, il faut utiliser les méthodes qui suivent

Ajouter, remplacer et supprimer des noeuds

- `node.removeChild(ancienNoeud)`
supprime l'enfant ancienNoeud du nœud node
- `node.appendChild(nouveauNoeud)`
ajoute nouveauNoeud en tant que nouveau (dernier) nœud enfant du nœud node
- `node.insertBefore(nouveauNoeud, ancienNoeud)`
insère nouveauNoeud en tant que nouveau nœud enfant du nœud node avant ancienNoeud
- `node.replaceChild(nouveauNoeud, ancienNoeud)`
remplace le nœud enfant ancienNoeud du nœud node par nouveauNoeud

Supprimer des noeuds

```
<div id="navigation">
  <ul id="nav">
    <li><strong>Home</strong></li>
    <li><a href="articles.html">Articles</a></li>
    <li><a href="photos.php" target="pop">Photos</a></li>
    <li><a href="clients.html">Clients</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</div>
```

Ajouter des noeuds

```
o=document.getElementById('nav');
p=document.createElement('li');
o.appendChild(p);

<div id="navigation">
  <ul id="nav">
    <li><strong>Home</strong></li>
    <li><a href="articles.html">Articles</a></li>
    <li><a href="photos.php" target="pop">Photos</a></li>
    <li><a href="clients.html">Clients</a></li>
    <li><a href="contact.html">Contact</a></li>
    <li></li>
  </ul>
</div>
```

Ajouter des nœuds avant un élément

```
o=document.getElementById('nav');
p=document.createElement('p');
p.appendChild(document.createTextNode('Menu:'));
o.parentNode.insertBefore(o,p);

<div id="navigation">
  <p>Menu: </p>
  <ul id="nav">
    <li><strong>Home</strong></li>
    <li><a href="articles.html">Articles</a></li>
    <li><a href="photos.php" target="pop">Photos</a></li>
    <li><a href="clients.html">Clients</a></li>
    <li><a href="contact.html">Contact </a></li>
    <li></li>
  </ul>
</div>
```

Remplacer des nœuds

```
o=document.getElementById('nav');
p=document.createElement('li');
p.appendChild(document.createTextNode('Adverts'));
o.replaceChild(p,o.lastChild);

<p>Menu: </p>
<ul id="nav">
  <li><strong>Home</strong></li>
  <li><a href="articles.html">Articles</a></li>
  <li><a href="photos.php" target="pop">Photos</a></li>
  <li><a href="clients.html">Clients</a></li>
  <li><a href="contact.html">Contact </a></li>
  <li>Adverts</li>
</ul>
</div>
```

Dupliquer des nœuds

```
o=document.getElementById('nav') ;
p=o.getElementsByTagName('li')[2] ;
dolly = p.cloneNode(true) ;
o.insertBefore(dolly,p) ;
```

```
<p>Menu: </p>
<ul id="nav">
  <li><strong>Home</strong></li>
  <li><a href="articles.html">Articles</a></li>
  <li><a href="photos.php" target="pop">Photos</a></li>
  <li><a href="photos.php" target="pop">Photos</a></li>
  <li><a href="clients.html">Clients</a></li>
  <li><a href="contact.html">Contact</a></li>
  <li>Adverts</li>
</ul>
</div>
```