

Chapitre 3

Sélectionner et manipuler le DOM

1. Sélectionner un élément unique

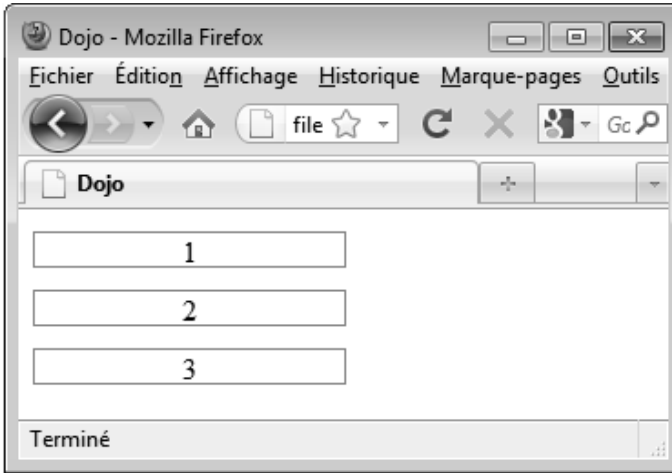
En JavaScript, la sélection d'un élément unique passe par son identifiant `id` et la méthode `document.getElementById`. Si la syntaxe de cette dernière est assez explicite, elle est plutôt longue à encoder. Le framework Dojo, comme la plupart des cadres JavaScript, propose une écriture raccourcie, en l'occurrence `dojo.byId`.

Exemple

Soit trois divisions.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Dojo</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-
8859-1">
<style type="text/css">
div { margin-top: 12px;
border: 1px solid gray;
width: 170px;
height: 18px;
text-align: center;}
</style>
</head>
<body>
```

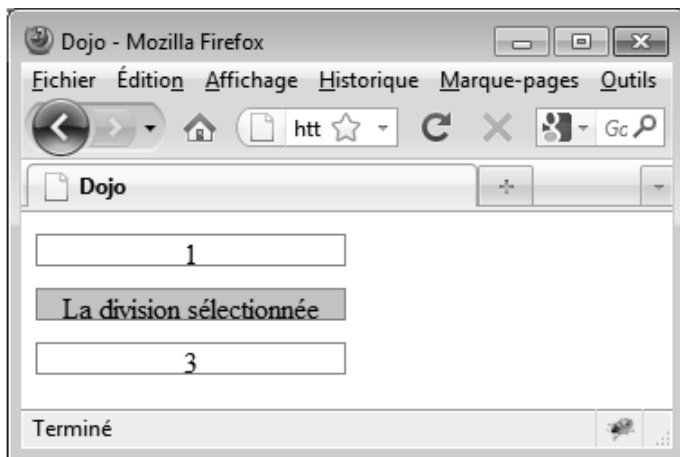
```
<div id="un">1</div>
<div id="deux">2</div>
<div id="trois">3</div>
</body>
</html>
```



Sélectionnons la division avec l'identifiant `id="deux"` au chargement de la page.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Dojo</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-
8859-1">
<style type="text/css">
div { margin-top: 12px;
border: 1px solid gray;
width: 170px;
height: 18px;
text-align: center;}
</style>
<script type="text/javascript" src="../js/dojo/dojo.js"
djConfig="parseOnLoad:true, isDebug: true">
</script>
<script type="text/javascript">
dojo.require("dojo.parser");
dojo.addOnLoad(function () {
```

```
dojo.byId("deux").innerHTML = "La division sélectionnée";
dojo.style("deux", {"backgroundColor" : "#9cf"});
});
</script>
</head>
<body>
<div id="un">1</div>
<div id="deux">2</div>
<div id="trois">3</div>
</body>
</html>
```



Détaillons ce script :

```
<script type="text/javascript" src="../../js/dojo/dojo.js"
djConfig="parseOnLoad:true, isDebug: true">
</script>
```

Chargement du framework Dojo.

```
<script type="text/javascript">
dojo.addOnLoad(function() {
...
});
```

Au chargement de la page (addOnLoad)...

```
dojo.byId("deux").innerHTML = "La division sélectionnée";
```

La division identifiée par deux est sélectionnée (`dojo.byId("deux")`) et son contenu est modifié par la propriété `innerHTML` du JavaScript classique (`innerHTML = "La division sélectionnée"`).

```
dojo.style("deux", {"backgroundColor" : "#9cf"});
```

Une couleur d'arrière-plan (`backgroundColor`) est appliquée à cette division deux par la fonction `dojo.style`.

2. Sélectionner des éléments multiples

Sélectionner un élément est intéressant mais il est parfois nécessaire de sélectionner un groupe d'éléments. C'est ce que permettra de faire `dojo.query`. Avec `dojo.query`, il sera possible d'intégrer, de façon compatible avec les navigateurs du marché, les sélecteurs CSS3.

Ce groupe d'éléments, éventuellement sélectionnés, est regroupé dans un tableau de type `array`.

Parcourons quelques sélecteurs CSS repris par `dojo.query`.

Les sélecteurs de base

#identifiant	Sélectionne un élément identifié par un id.	"#box"
.classe	Sélectionne les éléments possédant une classe déterminée.	".box"
balise	Sélectionne toutes les balises désignées.	"div"

Les sélecteurs hiérarchiques

A B	Sélectionne tout élément B qui est descendant d'un élément A.	"ul li"
A > B	Sélectionne tout élément B qui est un enfant direct d'un élément A.	"div > span"
A + B	Sélectionne tout élément B immédiatement précédé d'un élément A.	"ul#liste li"
A ~ B	Sélectionne tout élément B précédé d'un élément A.	"ul ~ li"

Les pseudo-classes de sélection

<code>:empty</code>	Correspond aux éléments vides et qui n'ont donc pas d'enfants. <code>"div:empty"</code>
<code>:checked</code>	Correspond aux éléments cochés dans les formulaires radio et de cases à cocher. <code>"input [type=checkbox] :checked"</code>
<code>:only-child</code>	Sélectionne le seul enfant. <code>div:only-child</code>
<code>:first-child</code>	Sélectionne le premier enfant. <code>ul:first-child</code>
<code>:last-child</code>	Sélectionne le dernier enfant. <code>ul:last-child</code>
<code>:nth-child(n)</code>	Désigne le nième élément enfant. <code>ul:nth-child(3)</code>
<code>:nth-child(even)</code>	Désigne les éléments enfants pairs. <code>ul:nth-child(even)</code>
<code>:nth-child(odd)</code>	Désigne les éléments enfants impairs. <code>ul:nth-child(odd)</code>
<code>:nth-last-child(n)</code>	Désigne le nième élément enfant en partant du dernier élément. <code>ul:nth-last-child(2)</code>
<code>:first-of-type</code>	Représente le premier élément de ce type. <code>li:first-of-type</code>
<code>:last-of-type</code>	Représente le dernier élément de ce type. <code>li:last-of-type</code>
<code>:nth-of-type(n)</code>	Représente le nième élément de ce type. <code>li:nth-of-type(3)</code>
<code>:nth-last-of-type(n)</code>	Représente le nième élément de ce type en partant du dernier élément. <code>li:nth-last-of-type(2)</code>

Les sélecteurs d'attributs

[attr]	Désigne l'élément qui comporte l'attribut attr indiqué. [class]
[attr="valeur"]	Désigne un élément qui comporte l'attribut attr fourni par la valeur indiquée. [class="classe1"]
[attr~="valeur"]	Correspond à tout élément dont l'attribut attr contient une liste de valeurs séparées par un espace et dont l'une d'elles est valeur. [class~="classe1"]
[attr ="valeur"]	Correspond à tout élément dont l'attribut attr contient une liste de valeurs séparées par des tirets débutant par valeur. [class ="classe1"]
[attr^="valeur"]	Représente un élément dont l'attribut attr commence avec le préfixe fourni par la valeur indiquée. [class^="cl"]
[attr\$="valeur"]	Représente un élément dont l'attribut attr se termine par le suffixe fourni par la valeur indiquée. [class\$="1"]
[attr*="valeur"]	Représente un élément dont l'attribut attr contient une instance de la valeur indiquée. [class*="las"]

Exemple 1

Sélectionnons l'élément de liste avec la classe class="deux".

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Dojo</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-
8859-1">
<style type="text/css">
li { width: 150px;}
</style>
<script type="text/javascript" src="../js/dojo/dojo.js"
```