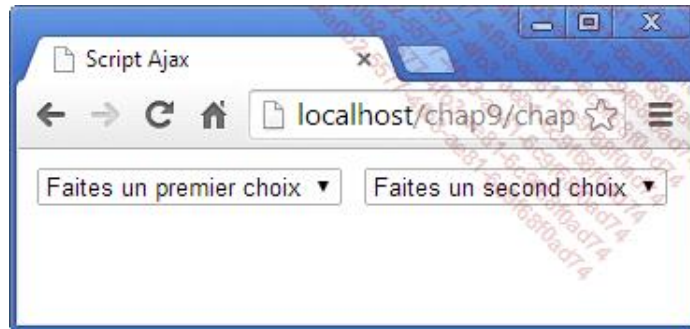


Un double menu déroulant

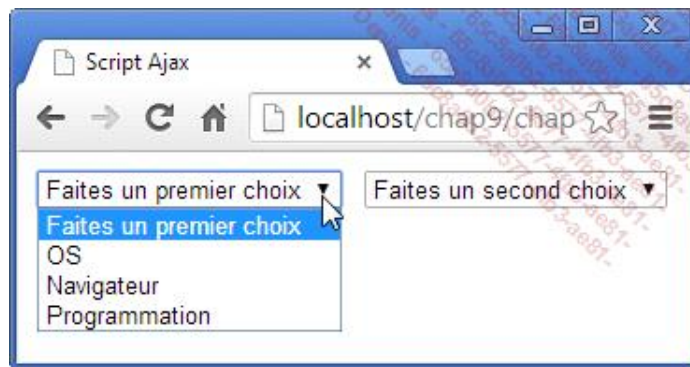
Cet exemple met en œuvre un double menu déroulant (balises `<select> ... </select>`) dans lequel, en fonction du choix effectué dans la première liste déroulante, les options de la seconde liste sont différentes, le tout sans faire appel à un langage serveur comme PHP.

Les captures d'écran suivantes illustrent le script.

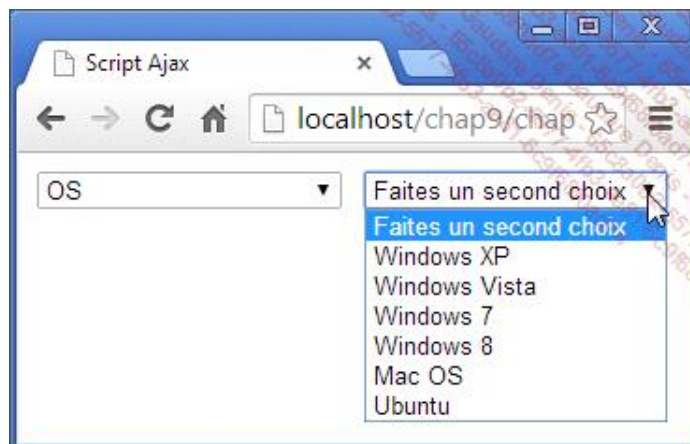
La situation de départ est la suivante :



Le premier menu propose les choix : OS, Navigateur et Programmation.

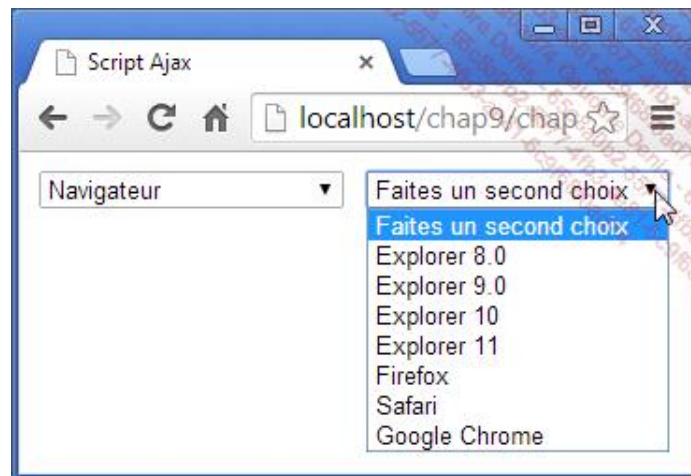


En choisissant l'option OS du premier menu, le second menu propose les options : Windows XP, Windows Vista, Windows 7, Windows 8, Mac OS et Ubuntu.

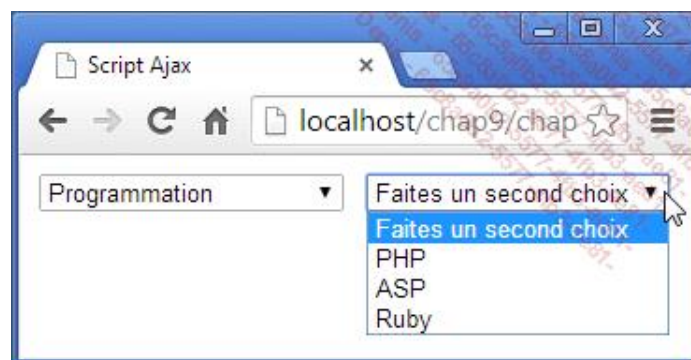


Si l'option Navigateur est retenue, les options deviennent : Explorer 8.0, Explorer 9.0, Explorer 10, Explorer 11,

Firefox, Safari et Google Chrome.



Et enfin pour le dernier choix Programmation, la seconde liste se change en : PHP, ASP et Ruby.



Concevons le fichier HTML de départ :

```
<!DOCTYPE html>
<html lang="fr">
<head>
<title>AJAX</title>
<meta charset="UTF-8">
</head>
<body>
<form name="formulaires">
<select name="form1" onChange="changement(this)">
<option>Faites un premier choix</option>
<option>OS</option>
<option>Navigateur</option>
<option>Programmation</option>
</select>&nbsp;
<select name="form2">
<option>Faites un second choix</option>
</select>
</form>
</body>
</html>
```

Commentaires :

- Le formulaire (global) porte le nom suivant : `name="formulaires"`.
- Les menus déroulants portent respectivement les noms `name="form1"` et `name="form2"`.
- Le choix effectué par l'utilisateur dans le premier menu déroulant est pris en charge par la fonction `changement (this)` du script.

Les éléments de la seconde balise `<select>` sont présents sur le serveur sous la forme d'un fichier texte comportant les différentes options séparées par le signe étoile (*).

Soit le fichier `OS.txt` :

```
*Windows XP*Windows Vista*Windows 7*Windows 8*Mac OS*Ubuntu
```

Le fichier `Navigateur.txt` :

```
*Explorer 8.0*Explorer 9.0*Explorer 10*Explorer 11*Firefox  
*Safari*Google Chrome
```

Et enfin, le fichier `Programmation.txt` :

```
*PHP*ASP*Ruby
```

Le script a pour mission, après avoir récupéré l'option sélectionnée au niveau du premier menu, de faire une requête sur le fichier texte concerné, de séparer les informations fournies par ce fichier et de les inclure dans le second menu.

Élaborons ce script, ligne par ligne :

```
<script>  
function changement(form1){
```

La fonction `changement ()` est appelée dans le fichier HTML lorsque l'utilisateur choisit une option dans le premier formulaire (`onChange="changement (this)"`).

```
var index = form1.selectedIndex;
```

La variable `index` identifie l'option sélectionnée par l'utilisateur dans le premier menu grâce à la propriété `selectedIndex`.

```
var valeur = form1[index].text;
```

La variable `valeur` retourne le texte associé à l'option (balise `<option> ... </option>`) sélectionnée.

```
var formulaireselements = document.getElementsByTagName("select");  
var form2elements = formulaireselements["form2"];
```

Plus loin dans le script, il faudra pointer les éléments du second menu. La variable `formulaireselements` liste tous les éléments du formulaire global (`formulaires`). La variable `form2elements` ne reprend que les éléments du second menu (`form2`).

```
url = valeur + ".txt";
```

Le nom du fichier texte à utiliser lors de la requête HTTP est construit à partir de la variable `valeur` à laquelle on ajoute l'extension ".txt".

```
if (valeur != "Faites votre choix") {
```

Lorsque l'utilisateur effectue un choix dans le premier menu déroulant, soit lorsque la variable `valeur` est différente de l'option proposée par défaut (`if (valeur != "Faites votre choix")`), la procédure pour initialiser la requête peut être démarrée.

```
var xhr = null;
if(window.XMLHttpRequest) {
xhr = new XMLHttpRequest();
}
else if(window.ActiveXObject){
xhr = new ActiveXObject("Microsoft.XMLHTTP");
}
else{
alert("Votre navigateur n'est pas compatible avec AJAX...");
}
```

L'objet `xhr` est créé de sorte à être compatible avec les différents navigateurs.

```
if(xhr) {
xhr.onreadystatechange = function(){
if(xhr.readyState == 4 && xhr.status == 200){
```

On s'assure que l'objet `xhr` a bien été créé (`if (xhr)`) et que la requête a bien abouti (`xhr.readyState == 4` et `xhr.status == 200`).

```
var form2reponse = xhr.responseText;
```

La variable `form2reponse` contient le fichier retourné par la requête sous forme d'un fichier texte (`xhr.responseText`).

```
contenuform2(form2reponse,form2elements);
}
}
xhr.open("GET", url, true);
xhr.send(null);
}
}
}
```

La fonction `contenuform2()`, qui va construire le second menu déroulant, peut alors être appelée, avec en arguments le fichier résultant de la requête (`form2reponse`) et les éléments du second menu (`form2elements`).

```
function contenuform2(form2reponse,form2elements) {
var form2options = form2reponse.split("*");
```

La première tâche de la fonction `contenuform2()` consiste à retrouver les différents items contenus dans le fichier `form2reponse`. Pour ce faire, la méthode `split()` de l'objet `String` est utilisée : elle retourne, sous forme de tableau, les éléments de la chaîne de caractères en tenant compte du séparateur étoile (*).

```
form2elements = 1;
form2elements.length = form2options.length;
```

On informe le moteur JavaScript que le nombre d'éléments du second menu (`form2elements.length`) sera égal au nombre de sous-chaînes de caractères contenues dans le tableau retourné par la méthode `split()` (`form2options.length`).

Par l'intermédiaire d'une boucle `for`, il suffit, pour terminer le script, de transférer une par une les sous-chaînes de caractères (`form2options[i]`) vers le texte associé aux options du second menu `<select>` (`form2elements[i].text`).

```
for (i=1; i < form2options.length; i++) {
form2elements[i].text = form2options[i];
}
}
//]]>
</script>
```

Le fichier HTML complet de cet exemple devient :

```
<!DOCTYPE html>
<html lang="fr">
<head>
<title>AJAX</title>
<meta charset="UTF-8">
<script>
function changement(form1){
var index = form1.selectedIndex;
var valeur = form1[index].text;
var formulaireelements = document.getElementsByTagName("select");
var form2elements = formulaireelements["form2"];
url = valeur + ".txt";
if (valeur != "Faites votre choix") {
var xhr = null;
if(window.XMLHttpRequest) {
xhr = new XMLHttpRequest();
}
else if(window.ActiveXObject){
xhr = new ActiveXObject("Microsoft.XMLHTTP");
}
else{
alert("Votre navigateur n'est pas compatible avec AJAX...");
}
if(xhr) {
xhr.onreadystatechange = function(){
```

```

if(xhr.readyState == 4 && xhr.status == 200){
var form2reponse = xhr.responseText;
contenuform2(form2reponse, form2elements);
}
}
xhr.open("GET", url, true);
xhr.send(null);
}
}
}

function contenuform2(form2reponse, form2elements) {
var form2options = form2reponse.split("*");
form2elements.length = 1;
form2elements.length = form2options.length;
for (i=1; i < form2options.length; i++) {
form2elements[i].text = form2options[i];
}
}
}
</script>
</head>
<body>
<form name="formulaires">
<select name="form1" onChange="changement(this)">
<option>Faites un premier choix</option>
<option>OS</option>
<option>Navigateur</option>
<option>Programmation</option>
</select>&nbsp;
<select name="form2">
<option>Faites un second choix</option>
</select>
</form>
</body>
</html>

```

Pour rendre le script fonctionnel, il faudrait associer une action (par exemple, un lien vers une page) à l'option retenue dans le second menu déroulant.