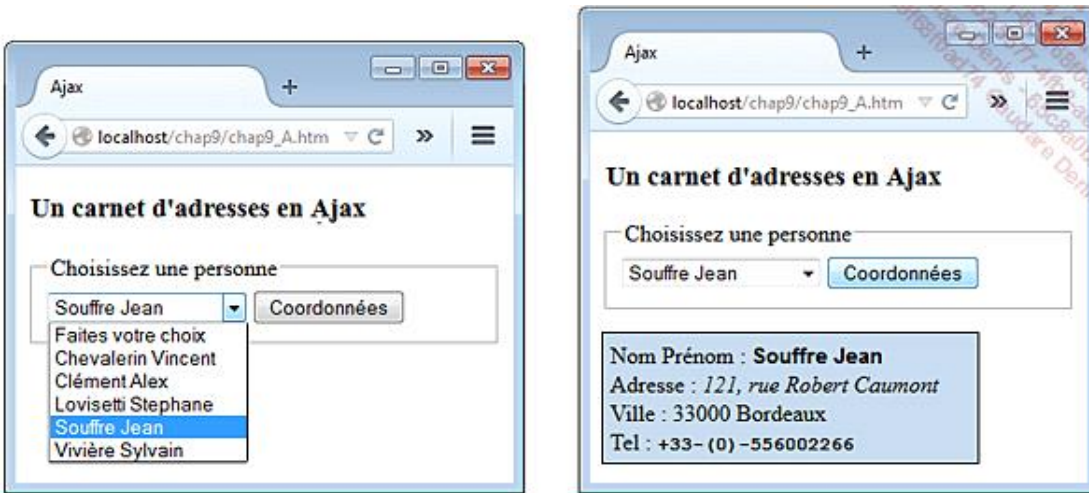


Une liste de contacts

Ce script propose une liste de sélection avec plusieurs noms. Au clic sur le bouton, les coordonnées de la personne sélectionnée sont affichées.

Les informations relatives aux personnes sont présentes sur le serveur sous forme d'un fichier XML.



Élaborons d'abord le fichier XML. Chaque contact du répertoire comporte les coordonnées suivantes :

- nom ;
- prénom ;
- adresse ;
- code postal ;
- ville ;
- téléphone.

La structure du fichier XML prend la forme suivante :

```
<carnet>
  <contact>
    <nom></nom>
    <prenom></prenom>
    <adresse></adresse>
    <cp></cp>
    <ville></ville>
    <tel></tel>
  </contact>
</carnet>
```

Le fichier XML (carnet.xml) proposé est :

```
<?xml version="1.0" encoding="UTF-8"?>
<carnet>
<contact>
```

```

<nom>Chevalerin</nom>
<prenom>Vincent</prenom>
<adresse>13, rue du Maréchal Joffre</adresse>
<cp>59000</cp>
<ville>Lille</ville>
<tel>+33-(0)-320219729</tel>
</contact>
<contact>
<nom>Clément</nom>
<prenom>Alex</prenom>
<adresse>17, boulevard Jean Jaurès</adresse>
<cp>31000</cp>
<ville>Toulouse</ville>
<tel>+33-(0)-565906248</tel>
</contact>
<contact>
<nom>Loviseti</nom>
<prenom>Stephane</prenom>
<adresse>9, avenue Poissonnière</adresse>
<cp>75000</cp>
<ville>Paris</ville>
<tel>+33-(0)-173409540</tel>
</contact>
<contact>
<nom>Souffre</nom>
<prenom>Jean</prenom>
<adresse>121, rue Robert Caumont</adresse>
<cp>33000</cp>
<ville>Bordeaux</ville>
<tel>+33-(0)-556002266</tel>
</contact>
<contact>
<nom>Vivière</nom>
<prenom>Sylvain</prenom>
<adresse>64, rue des Sources</adresse>
<cp>69000</cp>
<ville>Lyon</ville>
<tel>+33-(0)-478383899</tel>
</contact>
</carnet>

```

Passons maintenant au fichier HTML de départ :

```

<!DOCTYPE html>
<html lang="fr">
<head>
<title>AJAX</title>
<meta charset="UTF-8">
</head>
<body>
<h3>Un carnet d'adresses en Ajax</h3>
<form name="formulaire">
<fieldset>

```

```

<legend>Choisissez une personne</legend>
<select name="personnes">
<option>Faites votre choix</option>
<option>Chevalierin Vincent</option>
<option>Clément Alex</option>
<option>Loviseti Stephane</option>
<option>Souffre Jean</option>
<option>Vivière Sylvain</option>
</select>
<input type="button" onclick="extraire(personnes)"
value="Coordonnées">
</fieldset>
</form>
<div id="cadre" style="display:none;">
Nom Prénom : <span id="idnom"></span>&nbsp;<span
id="idprenom"></span>
<br>
Adresse : <span id="idadresse"></span><br>
Ville : <span id="idcp"></span>&nbsp;<span
id="idville"></span><br>
Tel : <span id="idtelephone"></span>
</div>
</body>
</html>

```

Commentaires :

- La liste de sélection est introduite par la balise `<select> ... </select>`.
- Les balises `<fieldset>...</fieldset>` et `<legend>...</legend>` sont des balises qui permettent d'organiser la présentation des formulaires.
- Le résultat s'affiche dans la zone `<div id="cadre"> ... </div>` qui, à l'ouverture de la page, est cachée (`style="display:none;"`).
- Au clic sur le bouton, la fonction `extraire(personnes)` permet de lancer le script.

Abordons la partie JavaScript.

```

<script>
var xhr = null;
function extraire(personnes){
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...");
}
}

```

Après avoir initialisé la variable `xhr`, élaborons le code de la fonction `extraire()`. Celle-ci est, pour rappel, appelée au clic sur le bouton **Coordonnées** dans le fichier HTML.

Les premières lignes de script tiennent compte des différences de traitement d'une requête HTTP d'un navigateur à l'autre. Google Chrome, Firefox, ainsi qu'Internet Explorer 7+ et les versions suivantes, passent par l'objet XMLHttpRequest, tandis qu'Internet Explorer 6 utilise un contrôle ActiveX.

```
if(xhr) {
xhr.onreadystatechange = function(){
if(xhr.readyState == 4 && xhr.status == 200){
var xmldocument = xhr.responseXML;
afficher(xmldocument,personnes);
}
}
xhr.open("GET", "carnet.xml", true);
xhr.send(null);
}
}
```

Après s'être assuré que l'objet `xhr` existe bien (`if(xhr)`), une requête est mise en place, selon la méthode GET, vers le fichier `carnet.xml` et en utilisant le mode asynchrone. La requête est alors effectivement exécutée lors de l'instruction `xhr.send(null)`.

La bonne exécution de cette requête HTTP est surveillée, en fonction des informations retournées sur ses propres changements d'état, par (`onreadystatechange`). L'assurance que la requête a bien abouti est obtenue à deux conditions : d'une part, que les données soient accessibles (`readyState == 4`), d'autre part, que le code retourné par le serveur soit satisfaisant (`status == 200`). Le fichier est alors retourné sous la forme d'un document XML (`xhr.responseXML`). Ce dernier est stocké dans la variable `xmldocument`. Le fichier XML peut maintenant être traité par le JavaScript du navigateur par la fonction `afficher()`. Le fichier XML (`xmldocument`) et les données du menu déroulant (`personnes`) sont fournis comme arguments de la fonction.

```
function afficher(xmldocument,personnes) {
```

La fonction `afficher()` se charge de récupérer les différentes coordonnées (nom, prénom, adresse, etc.) de la personne sélectionnée et d'afficher le résultat dans la page HTML.

```
var ordre = (personnes.selectedIndex - 1);
if(ordre!= -1) {
var affichage= document.getElementById("cadre");
affichage.style.display = 'block'
```

Il importe d'identifier la personne sélectionnée par l'utilisateur dans le menu déroulant. C'est le rôle de `selectedIndex`. Sa valeur est stockée dans la variable `ordre`. La première option du menu étant le texte "Faites votre choix", il faut prendre la précaution de diminuer cet index d'une unité (`selectedIndex - 1`). Tout affichage est exclu si cette valeur est négative (`if-(ordre!= -1)`). Le contenu de la balise `<div id="cadre"> ... </div>` devient apparent à ce stade du processus (`style.display = 'block'`), sachant qu'il est caché dans le fichier HTML initial.

```
noeudsnom = xmldocument.getElementsByTagName("nom");
noeudsprenom = xmldocument.getElementsByTagName("prenom");
noeudsadresse = xmldocument.getElementsByTagName("adresse");
noeudscp = xmldocument.getElementsByTagName("cp");
```

```
noeudsville = xmldocument.getElementsByTagName("ville");
noeudstel = xmldocument.getElementsByTagName("tel");
```

Les différentes balises (ex : <nom>) du fichier XML sont récupérées dans une liste (ex : noeuds.nom) grâce à la méthode `getElementsByTagName`. Les balises suivantes sont traitées de la même façon : <prenom>, <adresse>, <cp>, <ville> et <telephone>.

```
var textenom = noeudsnom[ordre].firstChild.nodeValue;
var texteprenom = noeudsprenom[ordre].firstChild.nodeValue;
var texteadresse = noeudsadresse[ordre].firstChild.nodeValue;
var textecp = noeudscp[ordre].firstChild.nodeValue;
var texteville = noeudsville[ordre].firstChild.nodeValue;
var textetel = noeudstel[ordre].firstChild.nodeValue;
```

Il faut maintenant atteindre le texte associé à la balise <nom> pour le contact sélectionné. Cette balise est déterminée par le numéro d'ordre du contact dans le menu déroulant (la variable `ordre`), associé à la liste `noeudsnom`. Dans l'arborescence du fichier XML, le nom du contact est le premier enfant de la balise <nom> (`firstChild`) ; la valeur de ce nœud texte est obtenue par la propriété `nodeValue`, et ainsi de suite pour les autres valeurs.

```
var targetnom = document.getElementById("idnom");
var targetprenom = document.getElementById("idprenom");
var targettrue = document.getElementById("idadresse");
var targetcp = document.getElementById("idcp");
var targetville = document.getElementById("idville");
var targettelephone = document.getElementById("idtelephone");
targetnom.innerHTML = textenom;
targetprenom.innerHTML = texteprenom;
targettrue.innerHTML = texteadresse;
targetcp.innerHTML = textecp;
targetville.innerHTML = texteville;
targettelephone.innerHTML = textetel;
}
}
```

La variable `targetnom` repère la balise ... du document HTML par son identifiant `id` (`getElementById("idnom")`). Le contenu de la balise est modifié par la propriété `innerHTML` (`targetnom.innerHTML`) en fonction des informations recueillies par la variable `textenom`.

```
</script>
```

Le script est terminé.

Sachant que la fonction `afficher()` doit être disponible avant la fonction `extraire()`, le script complet devient :

```
<script>
function afficher(xmldocument, personnes) {
var ordre = (personnes.selectedIndex - 1);
if(ordre!= -1) {
```

```

var affichage= document.getElementById("cadre");
affichage.style.display = 'block'
noeudsnom = xmldocument.getElementsByTagName("nom");
noeudsprenom = xmldocument.getElementsByTagName("prenom");
noeudsadresse = xmldocument.getElementsByTagName("adresse");
noeudscp = xmldocument.getElementsByTagName("cp");
noeudsville = xmldocument.getElementsByTagName("ville");
noeudstel = xmldocument.getElementsByTagName("tel");
var textenom = noeudsnom[ordre].firstChild.nodeValue;
var texteprenom = noeudsprenom[ordre].firstChild.nodeValue;
var texteadresse = noeudsadresse[ordre].firstChild.nodeValue;
var textecp = noeudscp[ordre].firstChild.nodeValue;
var texteville = noeudsville[ordre].firstChild.nodeValue;
var textetel = noeudstel[ordre].firstChild.nodeValue;
var targetnom = document.getElementById("idnom");
var targetprenom = document.getElementById("idprenom");
var targettrue = document.getElementById("idadresse");
var targetcp = document.getElementById("idcp");
var targetville = document.getElementById("idville");
var targettelephone = document.getElementById("idtelephone");
targetnom.innerHTML = textenom;
targetprenom.innerHTML = texteprenom;
targettrue.innerHTML = texteadresse;
targetcp.innerHTML = textecp;
targetville.innerHTML = texteville;
targettelephone.innerHTML = textetel;
}
}
var xhr = null;
function extraire(personnes){
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 xmldocument = xhr.responseXML;
afficher(xmldocument,personnes);
}
}
}
xhr.open("GET", "carnet.xml", true);
xhr.send(null);
}
}
</script>

```

Agrémentons le fichier HTML de quelques déclarations de style pour en améliorer la présentation.

```

<style>
#cadre { width: 250px;
        height: 80px;
        margin-top: 15px;
        padding: 5px;
        border: solid black 1px;
        background-color: rgb(195,215,235);}
#idnom, #idprenom { font-family: sans-serif;
                   font-weight: bold;
                   font-size: 11pt;}
#idadresse { font-style: italic;}
#idtelephone { font-weight: bold;
               font-family: monospace;}
</style>

```

Commentaires :

- Les différentes coordonnées (<div id="cadre">) sont affichées dans une boîte dotée d'une fine bordure et d'un arrière-plan de couleur. Ce dernier met en avant le résultat du script.
- Le nom et le prénom (et) sont affichés en gras.
- L'adresse () est mise en lettres italiques.
- Le téléphone () est présenté selon une police à chasse fixe.

Le fichier HTML complet, avec le script et les feuilles de style, devient :

```

<!DOCTYPE html>
<html lang="fr">
<head>
<title>AJAX</title>
<meta charset="UTF-8">
<script>
function afficher(xmldocument,personnes) {
var ordre = (personnes.selectedIndex - 1);
if(ordre!= -1) {
var affichage= document.getElementById("cadre");
affichage.style.display = 'block'
noeudsnom = xmldocument.getElementsByTagName("nom");
noeudsprenom = xmldocument.getElementsByTagName("prenom");
noeudsadresse = xmldocument.getElementsByTagName("adresse");
noeudscp = xmldocument.getElementsByTagName("cp");
noeudsville = xmldocument.getElementsByTagName("ville");
noeudstel = xmldocument.getElementsByTagName("tel");
var textenom = noeudsnom[ordre].firstChild.nodeValue;
var texteprenom = noeudsprenom[ordre].firstChild.nodeValue;
var texteadresse = noeudsadresse[ordre].firstChild.nodeValue;
var textecp = noeudscp[ordre].firstChild.nodeValue;
var texteville = noeudsville[ordre].firstChild.nodeValue;
var textetel = noeudstel[ordre].firstChild.nodeValue;
var targetnom = document.getElementById("idnom");
var targetprenom = document.getElementById("idprenom");
var targettrue = document.getElementById("idadresse");

```

```

var targetcp = document.getElementById("idcp");
var targetville = document.getElementById("idville");
var targettelephone = document.getElementById("idtelephone");
targetnom.innerHTML = textenom;
targetprenom.innerHTML = texteprenom;
targettrue.innerHTML = texteadresse;
targetcp.innerHTML = textecp;
targetville.innerHTML = texteville;
targettelephone.innerHTML = textetel;
}
}
var xhr = null;
function extraire(personnes){
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 xmldocument = xhr.responseXML;
afficher(xmldocument,personnes);
}
}
}
xhr.open("GET", "carnet.xml", true);
xhr.send(null);
}
}
</script>
<style>
#cadre { width: 250px;
height: 80px;
margin-top: 15px;
padding: 5px;
border: solid black 1px;
background-color: rgb(195,215,235);}
#idnom, #idprenom { font-family: sans-serif;
font-weight: bold;
font-size: 11pt;}
#idadresse { font-style: italic;}
#idtelephone { font-weight: bold;
font-family: monospace;}
</style>
</head>
<body>
<h3>Un carnet d'adresse en Ajax</h3>
<form action="" name="formulaire">
<fieldset>
<legend>Choisissez une personne</legend>

```



```
<select name="personnes">
<option>Faites votre choix</option>
<option>Chevalierin Vincent</option>
<option>Clément Alex</option>
<option>Lovisetti Stephane</option>
<option>Souffre Jean</option>
<option>Vivière Sylvain</option>
</select>
<input type="button" onclick="extraire(personnes)"
value="Coordonnées">
</fieldset>
</form>
<div id="cadre" style="display:none;">
Nom Prénom : <span id="idnom"></span>&nbsp;<span
id="idprenom"></span>
<br />
Adresse : <span id="idadresse"></span><br>
Ville : <span id="idcp"></span>&nbsp;<span
id="idville"></span><br>
Tel : <span id="idtelephone"></span>
</div>
</body>
</html>
```