

Ajax

Cours et notions

Acquisition de données depuis un serveur. Appel asynchrone et callback. Méthode Post ou Get.

Tous les exercices de cette section sont des adaptations relativement simples des exercices vus sur les données.

Affichage de données obtenues à partir d'un fichier XML ou JSON fourni

Appel AJAX, analyse du fichier reçu et affichage sous forme de liste ou de tableau.

Exercice 1 : un fichier de données XML est fourni, il faut en afficher le contenu

```
<!DOCTYPE html>
<html>
<head>
  <title>Test avec fichier XML</title>
  <meta content="text/html; charset=utf-8" http-equiv="Content-Type">
  <script type="text/javascript">
    var requeteAjax;
    function connect(page) {
      requeteAjax = new XMLHttpRequest();
      if (requeteAjax != null) {
        requeteAjax.open("GET", page, true);
        requeteAjax.onreadystatechange = handler;
        requeteAjax.send();
      }
      else {
        window.alert("Pas de support AJAX (XMLHTTP)");
      }
    }
    function handler() {
      if (requeteAjax.readyState === 4) {
        if (requeteAjax.status === 200) {
          display();
        }
      }
    }
    function call() {
      connect("donnees.xml");
    }
    function display() {
      var docXml = requeteAjax.responseXML;
      var reponses = docXml.getElementsByTagName("NOM");
      var texte = "";
      for (i = 0; i < reponses.length; i++) {
        texte = texte + reponses[i].firstChild.textContent + " ";
      }
      var divliste = document.getElementById("musiciens");
      divliste.style.visibility = "visible";
      divliste.value = texte;
    }
  </script>
</head>
<body>
  <form action="">
    <input id="Button1" type="button" value="Appeler" onclick="call()" />
  </form>
  <div id="data">
    <textarea id="musiciens" name="a" rows="50" cols="20"
style="visibility:hidden"></textarea>
  </div>
</body>
</html>
```

On pourra utiliser un fichier XML comme

```
<?xml version="1.0" encoding="UTF-8" ?>
<MUSICIENS>
  <MUSICIEN>
    <NOM>Bach</NOM>
    <PRENOM>Jean Sébastien</PRENOM>
  </MUSICIEN>
  <MUSICIEN>
    <NOM>Mozart</NOM>
    <PRENOM>Wolfgang Amadeus</PRENOM>
    <TEL> 01 45 23 67 89 </TEL>
  </MUSICIEN>
</MUSICIENS>
```

Exercice 2 : Même chose avec un fichier JSON fourni. L'analyse en est extrêmement simple :

```
var musiciens = JSON.parse(requeteAjax.responseText);
```

On pourra utiliser les données suivantes :

```
[
  { "Nom_Musicien": "Bacelli", "Prenom_Musicien": "Monica" },
  { "Nom_Musicien": "Bach", "Prenom_Musicien": "Carl-Philip-Emmanuel" },
  { "Nom_Musicien": "Bach", "Prenom_Musicien": "Johann Christian" }
]
```

Affichage des données reçues

Exercice 3 : Une page affichant les lettres de l'alphabet et chargeant toutes les données du fichier JSON mais n'affichant que les compositeurs correspondants par des appels Ajax.

Modification du panier réalisé à la séance précédente

Exercice 4 : Afficher les albums disponibles par un appel Ajax et réaliser un panier d'emprunt.