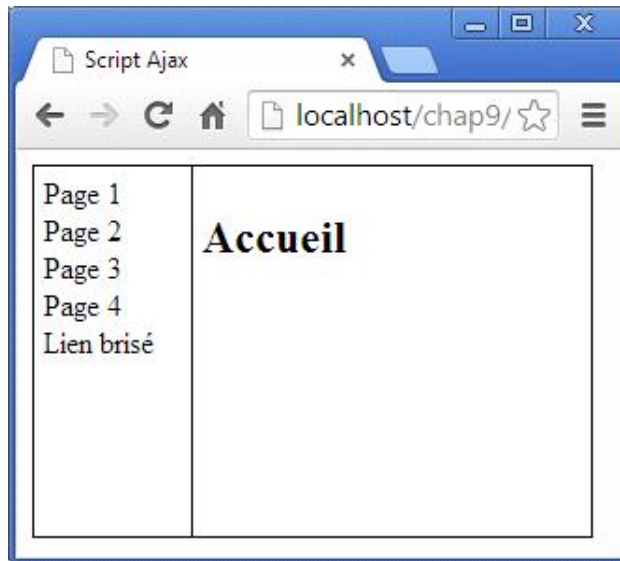


Un menu de navigation

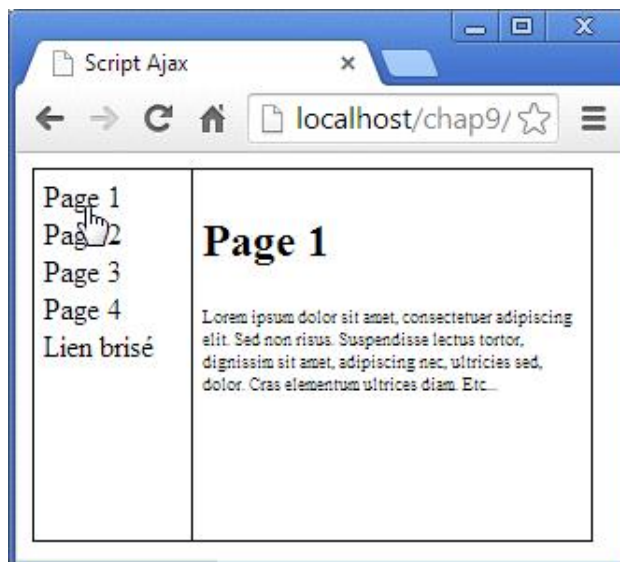
L'exemple suivant propose un tableau composé d'une ligne et de deux colonnes. En cliquant sur un lien du menu de navigation dans la première colonne, la page demandée s'affiche dans la seconde colonne.

Les pages affichées dans la seconde colonne sont présentes sur le serveur et sont appelées par une requête HTTP.

La capture d'écran illustre la situation d'origine :



Au clic sur le premier lien du menu de navigation, la page 1 s'affiche.



Le script prévoit la possibilité d'un lien brisé. Ce qui occasionne un message d'erreur illustré par la capture d'écran suivante :



Le fichier HTML de départ, avec quelques déclarations de style, se présente comme suit :

```
<!DOCTYPE html>
<html lang="fr">
<head>
<title>AJAX</title>
<meta charset="UTF-8">
<style>
#tableau { border-collapse: collapse;
            border: solid black 1px;
            height : 200px;}
.td1 { border-right : solid black 1px;
        padding: 5px;}
.td2 { padding: 5px;}
a:link { text-decoration:none; color:#000000;}
a:visited { text-decoration:none; color:#000000;}
a:hover { text-decoration:none; color:#000000;}
a:active { text-decoration:none; color:#000000;}
</style>
</head>
<body>
<table width="300" id="tableau">
<tr>
<td valign="top" width="80" class="td1">
<a href="javascript:void(0)" onclick="open_url('page1.htm')">Page
1</a><br>
<a href="javascript:void(0)" onclick="open_url('page2.htm')">Page
2</a><br>
<a href="javascript:void(0)" onclick="open_url('page3.htm')">Page
3</a><br>
<a href="javascript:void(0)" onclick="open_url('page4.htm')">Page
4</a><br>
<a href="javascript:void(0)" onclick="open_url('xxxxx.htm')">Lien
brisé</a>
</td>
<td valign="top" width="220" class="td2">
```

```

<div id="contenu"><h2>Accueil</h2></div>
</td>
</tr>
</table>
</body>
</html>

```

Le code de la balise de lien `<a> . . . ` nécessite quelques commentaires :

- Habituellement, au clic sur un lien, la page spécifiée dans l'attribut `href` est chargée par le navigateur. Dans ce script, ce traitement classique du HTML n'est pas utilisé, ainsi nous utilisons à la place la fonction `open_url ()`. Pour passer outre toute action de l'attribut `href`, la notation `javascript:void(0)` est utilisée.
- Au clic sur le lien, la fonction `open_url ()` est appelée et prend comme argument l'adresse du fichier à afficher, par exemple `open_url ('page1.htm')`.

Les fichiers `page1.htm`, `page2.htm`, `page3.htm` et `page4.htm` sont des fichiers HTML basiques. Voici le code du fichier `page1.htm` :

```

<!DOCTYPE html>
<html lang="fr">
<head>
<title>AJAX</title>
<meta charset="UTF-8">
</head>
<body>
<h2>Page 1</h2>
<p style="font-size: 9px">Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Sed non risus. Suspendisse lectus tortor,
dignissim sit amet, adipiscing nec, ultricies sed, dolor. Cras
elementum ultrices diam. Etc... </p>
</body>
</html>

```

Le script présente un codage dont la plupart des éléments devraient être assimilés à ce stade de l'étude.

```

<script>
var xhr = null;
function open_url(url) {

```

Après avoir défini la variable `xhr` comme une variable globale, la fonction `open_url(url)`, appelée au clic sur un lien dans le code HTML, est initialisée.

```

var xhr = null;
if (window.XMLHttpRequest) {
xhr = new XMLHttpRequest();
}
else {
if (window.ActiveXObject) {
try {
xhr = new ActiveXObject("Microsoft.XMLHTTP");

```

```
}  
catch (e) { }  
}  
}
```

L'objet `xhr` est créé pour être compatible avec les navigateurs Firefox et Internet Explorer.

```
xhr.onreadystatechange = function() { response(); }  
xhr.open("GET", url, true);  
xhr.send(null);  
}
```

La requête HTTP est effectuée. Aux changements d'états de celle-ci, la fonction `response()` est appelée.

```
function response() {  
    if (xhr.readyState == 4) {  
        if (xhr.status==200) {  
            document.getElementById("contenu").innerHTML = xhr.responseText;  
        }  
    }  
}
```

La fonction `response()`, après avoir contrôlé que les conditions habituelles sont remplies (`xhr.readyState == 4` et `xhr.status == 200`), va écrire dans la balise `<div id="contenu">` le fichier transmis sous forme d'un fichier texte. Il faut remarquer qu'il s'agit bien d'un fichier HTML car celui-ci répond bien à la définition de texte brut, agrémenté de balises permettant son affichage.

```
else {  
    document.getElementById("contenu").innerHTML = "<h3>La page n'est pas  
disponible. <br> <br> Erreur : " + xhr.status + "</h3>";  
}  
}  
}
```

Si la requête n'a pu aboutir à cause d'un lien brisé (si le fichier `xxxxxx.htm` n'est pas présent sur le serveur), un message d'erreur est affiché.

```
</script>
```

Fin du script.

Le fichier HTML complet, avec le script, se présente comme suit :

```
<!DOCTYPE html>  
<html lang="fr">  
<head>  
<title>AJAX</title>  
<meta charset="UTF-8">  
<script>  
var xhr = null;  
function open_url(url) {  
    if (window.XMLHttpRequest) {  
        xhr = new XMLHttpRequest();
```

```

}
else {
if (window.ActiveXObject) {
try {
xhr = new ActiveXObject("Microsoft.XMLHTTP");
}
catch(e) { }
}
}

xhr.onreadystatechange = function(){ reponse(); }
xhr.open("GET", url, true);
xhr.send(null);
}

function reponse() {
if (xhr.readyState == 4) {
if (xhr.status==200) {
document.getElementById("contenu").innerHTML = xhr.responseText;
}
else {
document.getElementById("contenu").innerHTML = "<h3>La page n'est
pas disponible.<br><br>Erreur : " + xhr.status + "</h3>";
}
}
}
</script>
<style>
#tableau { border-collapse: collapse;
            border: solid black 1px;
            height : 200px;}
.td1 { border-right : solid black 1px;
        padding: 5px;}
.td2 { padding: 5px;}
a:link { text-decoration:none; color:#000000;}
a:visited { text-decoration:none; color:#000000;}
a:hover { text-decoration:none; color:#000000;}
a:active { text-decoration:none; color:#000000;}
</style>
</head>
<body>
<table width="300" id="tableau">
<tr>
<td valign="top" width="80" class="td1">
<a href="javascript:void(0)" onclick="open_url('page1.htm')">Page
1</a><br>
<a href="javascript:void(0)" onclick="open_url('page2.htm')">Page
2</a><br>
<a href="javascript:void(0)" onclick="open_url('page3.htm')">Page
3</a><br>
<a href="javascript:void(0)" onclick="open_url('page4.htm')">Page
4</a><br>
<a href="javascript:void(0)" onclick="open_url('xxxxx.htm')">Lien
brisé</a>
</td>
<td valign="top" width="220" class="td2">
<div id="contenu"><h2>Accueil</h2></div>

```

```
</td>  
</tr>  
</table>  
</body>  
</html>
```