



- AJAX là viết tắt của Asynchronous JavaScript And XML.
- AJAX là một kiểu lập trình trở nên phổ biến vào năm 2005 (với Google Suggest).
- AJAX không phải là một ngôn ngữ lập trình mới mà là một cách thức mới sử dụng những chuẩn đã có.
- Với AJAX bạn có thể tạo ra những ứng dụng web tốt, nhanh và thân thiện với người dùng hơn.
- AJAX dựa trên những yêu cầu JavaScript và HTML.
- [Bắt đầu học ngay AJAX!](#)

Giới thiệu AJAX

AJAX viết tắt của Asynchronous JavaScript And XML.

Những yêu cầu kiến thức tối thiểu

Trước khi tiếp tục, bạn nên có những kiến thức cơ bản về:

- HTML, XHTML
- JavaScript

AJAX = Asynchronous JavaScript and XML

AJAX không là một ngôn ngữ lập trình mới, nhưng là một kỹ thuật để tạo ra những ứng dụng web tốt, nhanh và giao tiếp thân thiện hơn.

Với AJAX, JavaScript, bạn có thể giao tiếp trực tiếp với máy chủ bằng cách sử dụng đối tượng JavaScript là XMLHttpRequest. Với đối tượng này, JavaScript có thể trao đổi dữ liệu với máy chủ web mà không cần nạp lại trang.

AJAX sử dụng kỹ thuật chuyển dữ liệu bất đồng bộ (asynchronous) giữa trình duyệt và máy chủ web, cho phép các trang web yêu cầu thông tin từ máy chủ thay vì cả trang.

Công nghệ AJAX làm cho những ứng dụng Internet trở nên nhỏ, nhanh và thân thiện với người dùng hơn.

AJAX là một công nghệ phía trình duyệt, độc lập với phần mềm máy chủ web.

AJAX dựa trên các chuẩn về web

AJAX dựa trên các chuẩn về web sau đây:

- JavaScript
- XML
- HTML
- CSS

Các chuẩn về web được sử dụng trong AJAX đã được định nghĩa rất tốt, và được tất cả trình duyệt phổ biến hỗ trợ. Các ứng dụng AJAX đều độc lập với trình duyệt và nền.

AJAX làm cho những ứng dụng Internet tốt hơn

Những ứng dụng web có nhiều lợi ích hơn những ứng dụng máy để bàn; chúng có thể đến được với một số lượng lớn người dùng. Việc cài đặt, hỗ trợ và phát triển chúng dễ dàng hơn.

Tuy nhiên, những ứng dụng Internet không phải lúc nào cũng phong phú và thân thiện với người dùng như những ứng dụng truyền thống chạy trên máy để bàn.

Với AJAX, những ứng dụng web có thể phong phú và thân thiện với người dùng hơn.

Ngay bây giờ bạn có thể bắt đầu sử dụng AJAX

Không có gì mới để học.

AJAX dựa trên những chuẩn đã có. Những chuẩn này đã được các nhà phát triển sử dụng nhiều năm qua.

Những yêu cầu HTTP (HTTP Requests)

AJAX sử dụng những yêu cầu HTTP

Trong viết mã truyền thống, nếu bạn muốn lấy bất kỳ thông tin nào từ cơ sở dữ liệu, lấy tập tin trên máy chủ, hoặc gửi thông tin người dùng đến máy chủ, bạn sẽ phải tạo biểu mẫu HTML và GET hoặc POST dữ liệu đến máy chủ. Người dùng sẽ phải nhấp chuột vào nút "Submit" để gửi/nhận thông tin, chờ máy chủ hồi đáp, sau đó một trang mới sẽ được nạp với kết quả thu được.

Vì máy chủ trả về trang mới mỗi khi người dùng đồng ý gửi dữ liệu nhập, những ứng dụng web truyền thống có thể chạy một cách chậm chạp và trở nên thiếu thân thiện với người dùng.

Với AJAX, JavaScript của bạn giao tiếp trực tiếp với máy chủ, thông qua đối tượng JavaScript XMLHttpRequest.

Với một yêu cầu HTTP, trang web có thể tạo ra một yêu cầu và nhận phản hồi từ máy chủ web mà không cần nạp lại trang. Người dùng vẫn giữ nguyên trang cũ, và họ sẽ không bận tâm đến các mã kịch bản ngầm yêu cầu các trang hay ngầm gửi dữ liệu đến máy chủ.

Đối tượng XMLHttpRequest

Bằng cách sử dụng đối tượng XMLHttpRequest, nhà phát triển web có thể cập nhật trang với dữ liệu từ máy chủ sau khi trang đã được nạp.

AJAX được Google làm cho trở nên phổ biến vào năm 2005 (với Google Suggest).

Google Suggest sử dụng đối tượng XMLHttpRequest để tạo ra một giao diện rất linh hoạt: Khi bạn bắt đầu gõ chữ trong hộp tìm kiếm của Google, một JavaScript gửi những chữ đó đến máy chủ và máy chủ trả về một danh sách của những gợi ý.

Đối tượng XMLHttpRequest được hỗ trợ trong Internet Explorer 5.0+, Safari 1.2, Mozilla 1.0 / Firefox, Opera 8+, và Netscape 7.

Một ví dụ AJAX

Ứng dụng AJAX đầu tiên của bạn

Để hiểu AJAX làm việc thế nào, chúng ta sẽ tạo một ứng dụng AJAX nhỏ.

Trước tiên, chúng ta sẽ tạo một biểu mẫu HTML chuẩn với hai trường văn bản: *username* và *time*. Trường *username* sẽ được điền bởi người dùng và trường *time* sẽ được điền bằng cách sử dụng AJAX.

Tập tin HTML sẽ được đặt tên là "testAjax.htm", và nó như thế này (chú ý rằng biểu mẫu HTML bên dưới không có nút gửi dữ liệu!):

```
<html>
<body>

<form name="myForm">
Name: <input type="text" name="username" />
Time: <input type="text" name="time" />
</form>

</body>
</html>
```

Những phần kế tiếp sẽ giải thích trọng tâm của AJAX.

Việc hỗ trợ trình duyệt của AJAX

AJAX - Hỗ trợ trình duyệt

Trọng tâm của AJAX là đối tượng XMLHttpRequest.

Những trình duyệt khác nhau sử dụng những cách thức khác nhau để tạo đối tượng XMLHttpRequest.

Internet Explorer sử dụng ActiveXObject, trong khi những trình duyệt khác sử dụng đối tượng JavaScript dựng sẵn gọi là XMLHttpRequest.

Để tạo đối tượng này và để thích ứng với các trình duyệt khác nhau, chúng ta sẽ sử dụng lệnh *try* và *catch*. Bạn có thể tìm hiểu thêm về lệnh *try* và *catch* trong các tài liệu về JavaScript.

Hãy cập nhật tập tin "testAjax.htm" của bạn với JavaScript để tạo đối tượng XMLHttpRequest:

```
<html>
<body>

<script type="text/javascript">
function ajaxFunction()
{
    var xmlhttp;
    try
    {
        // Firefox, Opera 8.0+, Safari
        xmlhttp=new XMLHttpRequest();
    }
    catch (e)
    {
        // Internet Explorer
        try
        {
            xmlhttp=new ActiveXObject("Msxml2.XMLHTTP");
        }
        catch (e)
        {
            try
            {
                xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
            }
            catch (e)
            {
                alert("Your browser does not support AJAX!");
                return false;
            }
        }
    }
}
```

```
    }  
  }  
}  
}  
</script>  
  
<form name="myForm">  
Name: <input type="text" name="username" />  
Time: <input type="text" name="time" />  
</form>  
  
</body>  
</html>
```

Giải thích ví dụ: Trước tiên tạo biến xmlhttp để giữ đối tượng XMLHttpRequest.

Sau đó thử tạo đối tượng với xmlhttp = new XMLHttpRequest(). Đây là cho Firefox, Opera, và Safari browsers.

Nếu nó bị lỗi, thử xmlhttp = new ActiveXObject("Msxml2.XMLHTTP") dành cho Internet Explorer 6.0+, nếu nó vẫn bị lỗi.

Thử tiếp xmlhttp = new ActiveXObject("Microsoft.XMLHTTP") dành cho Internet Explorer 5.5+.

Nếu cả ba cách trên đều không làm việc, nghĩa người dùng đang có một trình duyệt quá cũ, và họ sẽ nhận được một thông báo về việc trình duyệt của họ không hỗ trợ AJAX.

Chú ý: Đoạn mã tương thích trình duyệt trên dài và khá phức tạp. Tuy nhiên, đây là đoạn mã bạn có thể sử dụng mỗi khi cần tạo một đối tượng XMLHttpRequest, khi ấy bạn chỉ cần chép và dán vào bất kỳ nơi nào bạn cần nó. Đoạn mã trên tương thích với tất cả trình duyệt phổ biến hiện nay như Internet Explorer, Opera, Firefox, và Safari.

Phần kế tiếp cho bạn thấy cách sử dụng đối tượng the XMLHttpRequest để giao tiếp với máy chủ.

AJAX - Đối tượng XMLHttpRequest

AJAX - Tìm hiểu thêm về đối tượng XMLHttpRequest

Trước khi gửi dữ liệu đến máy chủ, chúng ta phải hiểu rõ ba thuộc tính quan trọng của đối tượng XMLHttpRequest.

Thuộc tính onreadystatechange

Sau khi một yêu cầu gửi đến máy chủ, chúng ta cần một hàm để tiếp nhận dữ liệu được máy chủ trả về.

Thuộc tính onreadystatechange lưu giữ hàm này, hàm sẽ xử lý dữ liệu trả về từ máy chủ. Đoạn mã sau đây định nghĩa một hàm rỗng và đặt thuộc tính onreadystatechange cùng lúc:

```
xmlHttpRequest.onreadystatechange = function()
{
    // Chúng ta sẽ viết một vài dòng mã tại đây
}
```

Thuộc tính readyState

Thuộc tính readyState lưu giữ trạng thái phản hồi của máy chủ. Mỗi lần readyState thay đổi, hàm onreadystatechange sẽ được thi hành.

Đây là những giá trị hợp lệ cho thuộc tính readyState:

Trạng thái	Mô tả
0	Yêu cầu không được khởi động
1	Yêu cầu đã được cài đặt
2	Yêu cầu đã được gửi
3	Yêu cầu đang được xử lý
4	Yêu cầu đã hoàn tất

Chúng ta sẽ thêm một lệnh If vào hàm onreadystatechange để kiểm tra phản hồi của chúng ta đã hoàn tất hay chưa (có nghĩa là chúng ta có thể nhận dữ liệu của chúng ta):

```
xmlHttpRequest.onreadystatechange = function()
{
    if(xmlHttpRequest.readyState == 4)
    {
        // Lấy dữ liệu từ phản hồi của máy chủ
    }
}
```



```
}  
}
```

Thuộc tính responseText

Dữ liệu được gửi trả về từ máy chủ có thể được tiếp nhận với thuộc tính responseText.

Trong đoạn mã, chúng ta sẽ đặt giá trị của trường văn bản "time" bằng với responseText:

```
xmlHttp.onreadystatechange=function()  
{  
  if(xmlHttp.readyState==4)  
  {  
    document.myForm.time.value=xmlHttp.responseText;  
  }  
}
```

Phần kế tiếp trình bày cách yêu cầu máy chủ một vài dữ liệu!

AJAX - Yêu cầu máy chủ

AJAX - Gởi yêu cầu đến máy chủ

Để gởi ngấm một yêu cầu đến máy chủ, chúng ta sử dụng phương thức open() và send().

Phương thức open() nhận ba tham số. Tham số đầu tiên định nghĩa cách thức nào được sử dụng khi gởi yêu cầu (GET hoặc POST). Tham số thứ hai chỉ ra đường dẫn của tập tin mã kịch bản phía máy chủ. Tham số thứ ba chỉ ra yêu cầu có thực hiện bất đối xứng hay không. Phương thức send() gởi ngấm yêu cầu đến máy chủ. Nếu chúng ta giả sử rằng tập tin HTML và ASP ở cùng thư mục, đoạn mã như sau:

```
xmlHttp.open("GET", "time.asp", true);  
xmlHttp.send(null);
```

Bây giờ chúng ta phải quyết định khi nào hàm AJAX được thi hành. Chúng ta sẽ để hàm này chạy "bên dưới màn hình" khi người dùng gõ vài chữ trong trường văn bản username:

```
<form name="myForm">  
Name: <input type="text" onkeyup="ajaxFunction();" name="username" />  
Time: <input type="text" name="time" />  
</form>
```

Giờ đây tập tin đã cập nhật sẵn sàng cho AJAX "testAjax.htm" như sau:

```
<html>  
<body>  
  
<script type="text/javascript">  
function ajaxFunction()  
{  
    var xmlHttp;  
    try  
    {  
        // Firefox, Opera 8.0+, Safari  
        xmlHttp=new XMLHttpRequest();  
    }  
    catch (e)  
    {  
        // Internet Explorer  
        try  
        {  
            xmlHttp=new ActiveXObject("Msxml2.XMLHTTP");  
        }  
        catch (e)  
        {  

```

```
    try
    {
        xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");
    }
    catch (e)
    {
        alert("Your browser does not support AJAX!");
        return false;
    }
}

xmlHttp.onreadystatechange = function()
{
    if(xmlHttp.readyState == 4)
    {
        document.myForm.time.value = xmlHttp.responseText;
    }
}
xmlHttp.open("GET", "time.asp", true);
xmlHttp.send(null);
}
</script>

<form name="myForm">
Name: <input type="text"
onkeyup="ajaxFunction();" name="username" />
Time: <input type="text" name="time" />
</form>

</body>
</html>
```

Phần kế tiếp hoàn tất ứng dụng AJAX của chúng ta với tập tin mã kích bản "time.asp".

AJAX - Mã kịch bản phía máy chủ

AJAX - Mã kịch bản ASP phía máy chủ

Giờ đây chúng ta sẽ tạo mã kịch bản hiển thị thời gian hiện hành của máy chủ.

Thuộc tính `responseText` (đã giải thích trước đây) sẽ lưu giữ dữ liệu được trả về từ máy chủ. Ở đây chúng ta muốn gọi trở lại thời gian hiện hành. Đoạn mã trong "time.asp" như sau:

```
<%  
response.expires = -1  
response.write(time)  
%>
```

Chú ý: Thuộc tính `Expires` đặt thời lượng (theo phút) mà trang sẽ được lưu vùng nhớ tạm trên trình duyệt trước khi nó quá hạn. Nếu người dùng trở lại cùng trang trước khi nó quá hạn, phiên bản được lưu trong vùng nhớ tạm sẽ được hiển thị. `Response.Expires = -1` xác định trang sẽ không bao giờ được lưu tạm.

Chạy ứng dụng AJAX của bạn

Thử ứng dụng AJAX này bằng cách gõ vài chữ vào trong hộp văn bản *Name* bên dưới, sau đó nhấp chuột vào bên trong hộp văn bản *Time*:

Name: <input type="text"/>	Time: <input type="text"/>
----------------------------	----------------------------

Hộp văn bản *Time* nhận thời gian của máy chủ từ tập tin "time.asp" mà không cần nạp lại trang!

Ví dụ AJAX Suggest

Chúng ta vừa thấy AJAX được sử dụng để tạo ra các ứng dụng tương tác mạnh hơn.

Ví dụ AJAX Suggest

Trong ví dụ AJAX dưới đây sẽ minh họa cách một trang web có thể giao tiếp với máy chủ web trực tuyến ngay người dùng nhập dữ liệu vào trong biểu mẫu HTML.

Gõ một tên vào trong hộp dưới đây

First Name: <input type="text"/>
Suggestions:

Giải thích ví dụ - Biểu mẫu HTML

Biểu mẫu trên có mã HTML như sau:

```
<form>
First Name: <input type="text" id="txt1" onkeyup="showHint(this.value)">
</form>

<p>Suggestions: <span id="txtHint"></span></p>
```

Như bạn thấy nó chỉ là một biểu mẫu HTML đơn giản với một trường nhập được gọi là "txt1".

Thuộc tính sự kiện của trường này định nghĩa hàm được bẫy bởi sự kiện onkeyup.

Đoạn bên dưới biểu mẫu chứa một vùng được gọi là "txtHint". Vùng này được sử dụng như một nơi chứa dữ liệu được nhận về từ máy chủ web.

Khi người dùng nhập dữ liệu, một hàm gọi là "showHint()" được thi hành. Việc thi hành này được bẫy bởi sự kiện "onkeyup". Nói cách khác: Mỗi lần người dùng rời các ngón tay ra khỏi phím bên trong trường nhập, hàm showHint được gọi.

Giải thích ví dụ - Hàm showHint()

Hàm showHint() là một hàm JavaScript rất đơn giản được đặt trong khu vực <head> của trang HTML.

Hàm này chứa đoạn mã như sau:

```
function showHint(str)
{
  if (str.length==0)
  {
    document.getElementById("txtHint").innerHTML="";
    return;
  }
  xmlHttp=GetXmlHttpRequestObject();
  if (xmlHttp==null)
  {
    alert ("Your browser does not support AJAX!");
    return;
  }
  var url = "gethint.asp";
  url = url + "?q=" + str;
  url = url + "&sid=" + Math.random();
  xmlHttp.onreadystatechange = stateChanged;
  xmlHttp.open("GET",url,true);
  xmlHttp.send(null);
}
```

Hàm này thi hành mỗi khi một chữ được nhập vào trường nhập.

Nếu có vài chữ trong trường nhập ($str.length > 0$) hàm này thực hiện những việc sau:

Định nghĩa url (tên tập tin) để gọi đến máy chủ

Thêm một tham số (q) đến url với nội dung của trường nhập

Thêm một số ngẫu nhiên để tránh máy chủ sử dụng tập tin đệm

Tạo một đối tượng XMLHttpRequest, và bảo đối tượng thực hiện hàm gọi là stateChanged khi một thay đổi được xảy.

Mở đối tượng XMLHttpRequest với url đã cho.

Gửi một yêu cầu HTTP đến máy chủ

Nếu trường nhập rỗng, hàm đơn giản là xóa nội dung của nơi chứa txtHint.

Giải thích ví dụ - Hàm GetXmlHttpRequestObject()

Ví dụ trên gọi hàm GetXmlHttpRequestObject().

Mục đích của hàm này để giải quyết việc tạo các đối tượng XMLHttpRequest khác nhau cho các trình duyệt khác nhau.

Hàm này được liệt kê như sau:

```
function GetXmlHttpRequestObject()
{
    var xmlhttp=null;
    try
    {
        // Firefox, Opera 8.0+, Safari
        xmlhttp=new XMLHttpRequest();
    }
    catch (e)
    {
        // Internet Explorer
        try
        {
            xmlhttp=new ActiveXObject("Msxml2.XMLHTTP");
        }
        catch (e)
        {
            xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
        }
    }
    return xmlhttp;
}
```

Giải thích ví dụ - Hàm stateChanged()

Hàm stateChanged() chứa đoạn mã sau đây:

```
function stateChanged()
{
    if (xmlhttp.readyState==4)
    {
        document.getElementById("txtHint").innerHTML=xmlhttp.responseText;
    }
}
```

Hàm stateChanged() thực hiện mỗi khi trạng thái của đối tượng XMLHttpRequest thay đổi.

Khi trạng thái đổi thành 4 ("hoàn tất"), nội dung của nơi chứa txtHint được điền với văn bản phản hồi.

Mã nguồn của AJAX Suggest

Mã nguồn của ví dụ AJAX Suggest

Mã nguồn dưới đây đi theo ví dụ AJAX trong trang trước.

Bạn có thể chép và dán nó, và tự thử nó.

Trang HTML AJAX

Đây là trang HTML. Nó chứa một biểu mẫu HTML đơn giản và một liên kết đến một JavaScript.

```
<html>
<head>
  <script src="/clienthint.js"></script>
</head>
<body>
  <form>
    First Name: <input type="text" id="txt1"
                  onkeyup="showHint(this.value)">
  </form>
  <p>Suggestions: <span id="txtHint"></span></p>
</body>
</html>
```

Mã JavaScript được liệt kê bên dưới.

Mã AJAX JavaScript

Đây là mã JavaScript, lưu trữ trong tập tin "clienthint.js":

```
var xmlHttp

function showHint(str)
{
  if (str.length == 0)
  {
    document.getElementById("txtHint").innerHTML="";
    return;
  }
  xmlHttp = GetXmlHttpRequestObject();
  if (xmlHttp == null)
  {
    alert("Your browser does not support AJAX!");
    return;
  }
}
```



```

var url = "gethint.asp";
url = url + "?q=" + str;
url = url + "&sid=" + Math.random();
xmlHttp.onreadystatechange = stateChanged;
xmlHttp.open("GET",url,true);
xmlHttp.send(null);
}

function stateChanged()
{
    if (xmlHttp.readyState == 4)
    {
        document.getElementById("txtHint").innerHTML=xmlHttp.responseText;
    }
}

function GetXmlHttpRequestObject()
{
    var xmlHttp = null;
    try
    {
        // Firefox, Opera 8.0+, Safari
        xmlHttp=new XMLHttpRequest();
    }
    catch (e)
    {
        // Internet Explorer
        try
        {
            xmlHttp=new ActiveXObject("Msxml2.XMLHTTP");
        }
        catch (e)
        {
            xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");
        }
    }
    return xmlHttp;
}

```

Trang máy chủ AJAX - ASP và PHP

Không có gì nhiều về một máy chủ AJAX. Những trang AJAX có thể được lưu trữ bởi bất kỳ máy chủ Internet nào.

Trang máy chủ được gọi bởi JavaScript trong ví dụ từ chương trước là một tập tin ASP đơn giản gọi là "gethint.asp".

Dưới đây chúng ta liệt kê hai ví dụ của mã trang phía máy chủ, một được viết trong ASP và một được viết trong PHP.

Ví dụ ASP về AJAX

Đoạn mã trong trang "gethint.asp" được viết trong VBScript cho Internet Information Server (IIS). Nó chỉ kiểm tra một mảng của các tên và trả về những tên tương ứng đến máy khách:

```
<%  
response.expires = -1  
dim a(30)  
'Điền dữ liệu cho mảng với những cái tên  
a(1) = "Anna"  
a(2) = "Brittany"  
a(3) = "Cinderella"  
a(4) = "Diana"  
a(5) = "Eva"  
a(6) = "Fiona"  
a(7) = "Gunda"  
a(8) = "Hege"  
a(9) = "Inga"  
a(10) = "Johanna"  
a(11) = "Kitty"  
a(12) = "Linda"  
a(13) = "Nina"  
a(14) = "Ophelia"  
a(15) = "Petunia"  
a(16) = "Amanda"  
a(17) = "Raquel"  
a(18) = "Cindy"  
a(19) = "Doris"  
a(20) = "Eve"  
a(21) = "Evita"  
a(22) = "Sunniva"  
a(23) = "Tove"  
a(24) = "Unni"  
a(25) = "Violet"  
a(26) = "Liza"  
a(27) = "Elizabeth"  
a(28) = "Ellen"  
a(29) = "Wenche"  
a(30) = "Vicky"  
  
'lấy tham số q từ URL  
q = ucase(request.querystring("q"))  
  
'tra tất cả gợi ý từ mảng nếu độ dài của q > 0  
if len(q) > 0 then  
    hint = ""  
    for i=1 to 30  
        if q = ucase(mid(a(i),1,len(q))) then  
            if hint = "" then
```

```

        hint = a(i)
    else
        hint = hint & " , " & a(i)
    end if
end if
next
end if

'xuất "no suggestion" nếu không tìm thấy gợi ý nào hoặc xuất những giá
trị đúng
if hint = "" then
    response.write("no suggestion")
else
    response.write(hint)
end if
%>

```

Ví dụ PHP về AJAX

Đoạn mã sau đây được viết trong PHP.

Chú ý: Để chạy toàn bộ ví dụ trong PHP, nhớ đổi giá trị của biến đường dẫn trong "clienthint.js" từ "gethint.asp" thành "gethint.php".

Ví dụ PHP

```

<?php
header("Cache-Control: no-cache, must-revalidate");

// Date in the past
header("Expires: Mon, 26 Jul 1997 05:00:00 GMT");

// Fill up array with names
$a[] = "Anna";
$a[] = "Brittany";
$a[] = "Cinderella";
$a[] = "Diana";
$a[] = "Eva";
$a[] = "Fiona";
$a[] = "Gunda";
$a[] = "Hege";
$a[] = "Inga";
$a[] = "Johanna";
$a[] = "Kitty";
$a[] = "Linda";
$a[] = "Nina";
$a[] = "Ophelia";
$a[] = "Petunia";
$a[] = "Amanda";
$a[] = "Raquel";

```

```

$a[] = "Cindy";
$a[] = "Doris";
$a[] = "Eve";
$a[] = "Evita";
$a[] = "Sunniva";
$a[] = "Tove";
$a[] = "Unni";
$a[] = "Violet";
$a[] = "Liza";
$a[] = "Elizabeth";
$a[] = "Ellen";
$a[] = "Wenche";
$a[] = "Vicky";

//lấy tham số q từ URL
$q = $_GET["q"];

//tra tất cả gợi ý từ mảng nếu độ dài của q > 0
if (strlen($q) > 0)
{
    $hint = "";
    for($i=0; $i<count($a); $i++)
    {
        if (strtolower($q) == strtolower(substr($a[$i],0,strlen($q)))
        {
            if ($hint == "")
            {
                $hint = $a[$i];
            }
            else
            {
                $hint = $hint . " , " . $a[$i];
            }
        }
    }
}

// xuất ra "no suggestion" nếu không tìm thấy gợi ý hoặc giá trị đúng
if ($hint == "")
{
    $response = "no suggestion";
}
else
{
    $response = $hint;
}

//xuất phản hồi
echo $response;
?>

```

AJAX Database Example

AJAX có thể được sử dụng cho giao tiếp với cơ sở dữ liệu.

Ví dụ Cơ sở dữ liệu AJAX Database

Trong ví dụ AJAX bên dưới chúng ta sẽ minh họa cách một trang web có thể lấy thông tin từ một cơ sở dữ liệu bằng công nghệ AJAX.

Chọn một tên trong hộp bên dưới

Select a Customer:

Customer info will be listed here.

Giải thích ví dụ AJAX

Ví dụ trên chứa một biểu mẫu HTML đơn giản và một liên kết đến JavaScript:

```
<html>
<head>
<script src="selectcustomer.js"></script>
</head>
<body>
<form>
Select a Customer:
<select name="customers" onchange="showCustomer(this.value)">
<option value="ALFKI">Alfreds Futterkiste
<option value="NORTS ">North/South
<option value="WOLZA">Wolski Zajazd
</select>
</form>
<p>
<div id="txtHint"><b>Customer info will be listed here.</b></div>
</p>
</body>
</html>
```

Như bạn thấy nó chỉ là một biểu mẫu HTML đơn giản với một hộp danh sách xổ xuống gọi là "customers".

Đoạn bên dưới biểu mẫu chứa một vùng gọi là "txtHint". Vùng này được dùng làm nơi chứa thông tin nhận về từ máy chủ web.

Khi người dùng chọn dữ liệu, hàm "showCustomer()" được thi hành. Việc thi hành của hàm này được bắt bởi sự kiện "onchange". Nói cách khác: Mỗi khi người dùng thay đổi giá trị trong hộp xổ xuống, hàm showCustomer được gọi.

Mã JavaScript được liệt kê bên dưới.

Mã JavaScript AJAX

Đây là mã JavaScript chứa trong tập tin "selectcustomer.js":

```
var xmlhttp

function showCustomer(str)
{
    xmlhttp=GetXmlHttpRequest();
    if (xmlhttp==null)
    {
        alert ("Your browser does not support AJAX!");
        return;
    }
    var url = "getcustomer.asp";
    url = url+ "?q=" + str;
    url = url + "&sid=" + Math.random();
    xmlhttp.onreadystatechange = stateChanged;
    xmlhttp.open("GET",url,true);
    xmlhttp.send(null);
}

function stateChanged()
{
    if (xmlhttp.readyState == 4)
    {
        document.getElementById("txtHint").innerHTML = xmlhttp.responseText;
    }
}

function GetXmlHttpRequest()
{
    var xmlhttp = null;
    try
    {
        // Firefox, Opera 8.0+, Safari
        xmlhttp = new XMLHttpRequest();
    }
    catch (e)
    {
        // Internet Explorer
        try
        {
            xmlhttp=new ActiveXObject("Msxml2.XMLHTTP");
```

```
    }
    catch (e)
    {
        xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");
    }
}
return xmlHttp;
}
```

Trang máy chủ AJAX

Trang máy chủ được gọi bởi JavaScript, là một tập tin ASP đơn giản gọi là "getcustomer.asp".

Trang này được viết trong VBScript cho Internet Information Server (IIS). Nó cũng được viết dễ dàng trong PHP, hoặc một vài ngôn ngữ khác.

Mã này chạy một SQL đến cơ sở dữ liệu và trả về kết quả như một bảng biểu HTML:

```
<%
response.expires = -1
sql = "SELECT * FROM CUSTOMERS WHERE CUSTOMERID="
sql = sql & "'" & request.querystring("q") & "'"

set conn = Server.CreateObject("ADODB.Connection")
conn.Provider = "Microsoft.Jet.OLEDB.4.0"
conn.Open(Server.MapPath("/db/northwind.mdb"))
set rs = Server.CreateObject("ADODB.recordset")
rs.Open sql, conn

response.write("<table>")
do until rs.EOF
    for each x in rs.Fields
        response.write("<tr><td><b>" & x.name & "</b></td>")
        response.write("<td>" & x.value & "</td></tr>")
    next
    rs.MoveNext
loop

response.write("</table>")
%>
```

Ví dụ AJAX XML

AJAX có thể được dùng để giao tiếp với tập tin XML.

Ví dụ AJAX XML

Trong ví dụ AJAX bên dưới chúng ta sẽ minh họa cách một trang web có thể lấy thông tin từ một tập tin XML bằng công nghệ AJAX.

Chọn một CD trong hộp bên dưới

Select a CD:

CD info will be listed here.

Giải thích ví dụ

Ví dụ trên chứa một biểu mẫu HTML đơn giản và một liên kết đến JavaScript:

```
<html>
<head>
<script src="selectcd.js"></script>
</head>
<body>
<form>
Select a CD:
<select name="cds" onchange="showCD(this.value)">
<option value="Bob Dylan">Bob Dylan</option>
<option value="Bonnie Tyler">Bonnie Tyler</option>
<option value="Dolly Parton">Dolly Parton</option>
</select>
</form>
<p>
<div id="txtHint"><b>CD info will be listed here.</b></div>
</p>
</body>
</html>
```

Bạn thấy nó chỉ là một biểu mẫu HTML đơn giản và một hộp xổ xuống gọi là "cds".

Đoạn văn bản bên dưới biểu mẫu chứa một vùng gọi là "txtHint". Vùng này được dùng lưu trữ thông tin tiếp nhận từ máy chủ..

Khi người dùng chọn dữ liệu, hàm "showCD" được thi hành. Việc thi hành của hàm này được bắt bởi sự kiện "onchange". Nói cách khác: Mỗi khi người dùng thay đổi giá trị trong hộp xổ xuống, hàm showCD được gọi.

Mã JavaScript được liệt kê bên dưới.

Mã JavaScript AJAX

Đây là mã JavaScript chứa trong tập tin "selected.js":

```
var xmlHttp
function showCD(str)
{
    xmlHttp = GetXmlHttpRequestObject();
    if (xmlHttp == null)
    {
        alert ("Your browser does not support AJAX!");
        return;
    }
    var url = "getcd.asp";
    url = url + "?q=" + str;
    url = url + "&sid=" + Math.random();
    xmlHttp.onreadystatechange = stateChanged;
    xmlHttp.open("GET",url,true);
    xmlHttp.send(null);
}

function stateChanged()
{
    if (xmlHttp.readyState == 4)
    {
        document.getElementById("txtHint").innerHTML = xmlHttp.responseText;
    }
}

function GetXmlHttpRequestObject()
{
    var xmlHttp = null;
    try
    {
        // Firefox, Opera 8.0+, Safari
        xmlHttp = new XMLHttpRequest();
    }
    catch (e)
    {
        // Internet Explorer
        try
        {
            xmlHttp = new ActiveXObject("Msxml2.XMLHTTP");
        }
        catch (e)
        {
            xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");
        }
    }
}
```

```
}  
    return xmlHttp;  
}
```

Trang máy chủ AJAX

Trang máy chủ được gọi bởi JavaScript, là một tập tin ASP đơn giản gọi là "getcd.asp".

Trang này được viết trong VBScript cho Internet Information Server (IIS). Nó cũng được viết dễ dàng trong PHP, hoặc một vài ngôn ngữ khác.

Mã này chạy một truy vấn đến tập XML và trả về kết quả dạng HTML:

```
<%  
response.expires = -1  
q=request.querystring("q")  
  
set xmlDoc = Server.CreateObject("Microsoft.XMLDOM")  
xmlDoc.async = "false"  
xmlDoc.load(Server.MapPath("cd_catalog.xml"))  
  
set nodes = xmlDoc.selectNodes("CATALOG/CD[ARTIST='" & q & "'"]")  
  
for each x in nodes  
    for each y in x.childnodes  
        response.write("<b>" & y.nodename & ":</b> ")  
        response.write(y.text)  
        response.write("<br />")  
    next  
next  
%>
```

Ví dụ về hỏi đáp XML AJAX

Trong khi `responseText` trả về phản hồi HTTP như một chuỗi, `responseXML` trả về phản hồi như một XML.

Thuộc tính `ResponseXML` trả về một đối tượng tài liệu XML, nó có thể được nhận diện và tách bằng cách dùng các phương thức và thuộc tính của cấu trúc cây W3C DOM.

Ví dụ hỏi đáp AJAX ResponseXML

Trong ví dụ AJAX sau đây chúng ta sẽ minh họa cách một trang có thể lấy thông tin từ một cơ sở dữ liệu bằng công nghệ AJAX. Những dữ liệu được chọn từ cơ sở dữ liệu lần này sẽ được chuyển đổi thành một tài liệu XML, và sau đó chúng ta sẽ sử dụng DOM để kết xuất những giá trị để hiển thị.

Chọn tên trong hộp dưới đây

Select a Customer:

Giải thích ví dụ AJAX

Ví dụ trên chứa một biểu mẫu HTML, các phân tử `` lưu giữ các dữ liệu trả về, và một liên kết đến JavaScript:

```
<html>
<head>
<script src="selectcustomer_xml.js"></script>
</head>
<body>
<form action="">
Select a Customer:
<select name="customers" onchange="showCustomer(this.value)">
<option value="ALFKI">Alfreds Futterkiste</option>
<option value="NORTS ">North/South</option>
<option value="WOLZA">Wolski Zajazd</option>
</select>
</form>
<b><span id="companyname"></span></b><br />
<span id="contactname"></span><br />
<span id="address"></span>
<span id="city"></span><br/>
<span id="country"></span>
</body>
</html>
```

Ví dụ trên chứa một biểu mẫu HTML và một hộp xổ xuống gọi là "customers".

Khi người dùng chọn một khách hàng trong hộp xổ xuống, hàm "showCustomer()" được thi hành. Việc thi hành của hàm này được bẫy bởi sự kiện "onchange". Nói cách khác: Mỗi khi người dùng thay đổi giá trị trong hộp xổ xuống, hàm showCustomer() được gọi.

Mã JavaScript được liệt kê bên dưới.

Mã AJAX JavaScript

Đây là mã JavaScript lưu trữ trong tập tin "selectcustomer_xml.js":

```
var xmlHttp
function showCustomer(str)
{
    xmlHttp=GetXmlHttpObject();
    if (xmlHttp==null)
    {
        alert ("Your browser does not support AJAX!");
        return;
    }
    var url = "getcustomer_xml.asp";
    url = url + "?q=" + str;
    url = url + "&sid=" + Math.random();
    xmlHttp.onreadystatechange = stateChanged;
    xmlHttp.open("GET",url,true);
    xmlHttp.send(null);
}

function stateChanged()
{
    if (xmlHttp.readyState==4)
    {
        var xmlDoc = xmlHttp.responseXML.documentElement;
        document.getElementById("companyname").innerHTML=
xmlDoc.getElementsByTagName("compname")[0].childNodes[0].nodeValue;
        document.getElementById("contactname").innerHTML=
xmlDoc.getElementsByTagName("contname")[0].childNodes[0].nodeValue;
        document.getElementById("address").innerHTML=
xmlDoc.getElementsByTagName("address")[0].childNodes[0].nodeValue;
        document.getElementById("city").innerHTML=
xmlDoc.getElementsByTagName("city")[0].childNodes[0].nodeValue;
        document.getElementById("country").innerHTML=
xmlDoc.getElementsByTagName("country")[0].childNodes[0].nodeValue;
    }
}

function GetXmlHttpObject()
{
```

```

var xmlHttp=null;
try
{
    // Firefox, Opera 8.0+, Safari
    xmlHttp = new XMLHttpRequest();
}
catch (e)
{
    // Internet Explorer
    try
    {
        xmlHttp = new ActiveXObject("Msxml2.XMLHTTP");
    }
    catch (e)
    {
        xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");
    }
}
return xmlHttp;
}

```

Các hàm showCustomer() và GetXmlHttpRequest() trên giống như các phần trước. Hàm stateChanged() cũng đã được đề cập trước đây, tuy nhiên; lần này chúng ta trả về kết quả là tài liệu XML (với responseXML) và sử dụng DOM để kết xuất các giá trị để hiển thị.

Trang máy chủ AJAX

Trang máy chủ được gọi bởi JavaScript, là một tập tin ASP đơn giản gọi là "getcustomer_xml.asp".

Trang này được viết trong VBScript cho Internet Information Server (IIS). Nó cũng được viết dễ dàng trong PHP, hoặc một vài ngôn ngữ khác.

Mã này chạy một truy vấn SQL đến cơ sở dữ liệu và trả về kết quả là một tài liệu XML:

```

<%
response.expires = -1
response.contentType = "text/xml"
sql = "SELECT * FROM CUSTOMERS "
sql = sql & " WHERE CUSTOMERID='" & request.querystring("q") & "'"

on error resume next
set conn = Server.CreateObject("ADODB.Connection")
conn.Provider = "Microsoft.Jet.OLEDB.4.0"
conn.Open(Server.MapPath("/db/northwind.mdb"))
set rs = Server.CreateObject("ADODB.recordset")
rs.Open sql, conn
if err <> 0 then

```

```
response.write(err.description)
set rs = nothing
set conn = nothing
else
response.write("<?xml version='1.0' encoding='ISO-8859-1'?>")
response.write("<company>")
response.write("<compname>" &rs.fields("companyname")& "</compname>")
response.write("<contname>" &rs.fields("contactname")& "</contname>")
response.write("<address>" &rs.fields("address")& "</address>")
response.write("<city>" &rs.fields("city")& "</city>")
response.write("<country>" &rs.fields("country")& "</country>")
response.write("</company>")
end if
on error goto 0
%>
```

Đề ý dòng thứ hai trong mã ASP trên: `response.contentType="text/xml"`. Thuộc tính `ContentType` đặt kiểu nội dung HTTP cho đối tượng hồi đáp. Giá trị mặc định cho thuộc tính này là `"text/html"`. Lần này chúng ta muốn kiểu nội dung là XML.

Sau khi chọn dữ liệu từ cơ sở dữ liệu, chúng ta xây dựng một tài liệu XML với dữ liệu nhận được.