

A stylized graphic of a globe with a white landmass silhouette on a grey circular background. The globe is centered and has a thick grey arc passing behind it. Two orange lines, one horizontal and one vertical, intersect at the center of the globe, forming a crosshair.

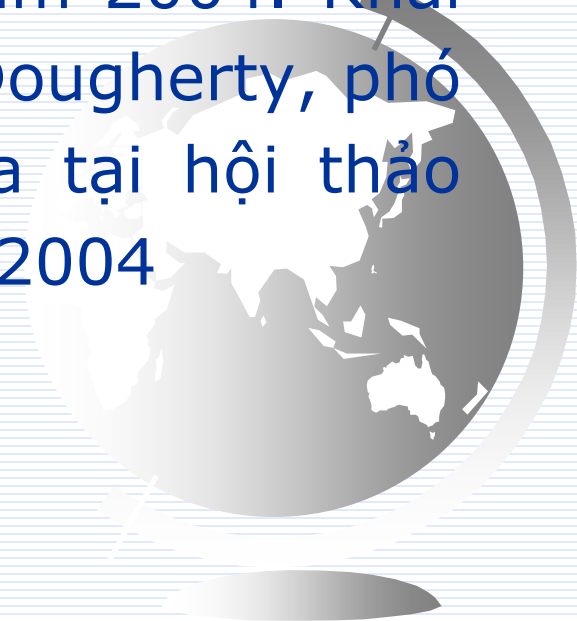
PHP và lập trình web 2.0

WEB 1.0

- ✓ World Wide Web, gọi tắt là Web hoặc WWW, mạng lưới toàn cầu là một không gian thông tin toàn cầu mà mọi người có thể truy nhập qua các máy tính nối với mạng Internet.
- ✓ Thuật ngữ này thường bị hiểu nhầm là từ đồng nghĩa với chính thuật ngữ Internet. Nhưng Web thực ra chỉ là một trong các dịch vụ chạy trên Internet, chẳng hạn như dịch vụ thư điện tử. Web được phát minh và đưa vào sử dụng vào khoảng năm 1990, 1991 bởi viện sĩ Viện Hàn lâm Anh Tim Berners-Lee và Robert Cailliau (Bỉ) tại CERN, Geneva, Switzerland

WEB 2.0

- ✓ Web 2.0: là một xu hướng trong thiết kế và phát triển web mà mục đích là nhằm tới những thuận tiện trong việc chia sẻ, hợp tác và sáng tạo giữa các người dùng.
- ✓ Thuật ngữ được sử dụng lần đầu tiên trong buổi hội thảo O'Reilly Media Web 2.0 năm 2004. Khái niệm **Web 2.0** đầu tiên được Dale Dougherty, phó chủ tịch của O'Reilly Media, đưa ra tại hội thảo Web 2.0 lần thứ nhất vào tháng 10/2004.



Web 1.0 và Web 2.0

- ✓ Dougherty không đưa ra định nghĩa mà chỉ dùng các ví dụ so sánh phân biệt Web 1.0 và Web 2.0:

Web 1.0	Web 2.0
<i>DoubleClick.com</i> <i>Ofoto.com</i> ...	<i>Google.com</i> <i>Flickr.com</i> ...



✓ Một số công nghệ:

- XML

- **AJAX**

- RSS

- ...



AJAX

- ✓ AJAX = **A**synchronous **J**avaScript **A**nd **X**ML
- ✓ AJAX không phải là ngôn ngữ mới mà là một cách mới sử dụng các ngôn ngữ đã có.
- ✓ AJAX giúp ta tạo các trang web nhanh hơn, tiện lợi hơn, thân thiện hơn khi sử dụng.
- ✓ AJAX dựa trên JavaScript và các yêu cầu HTTP.



- ✓ Là công nghệ của web browser. Độc lập với web server.
- ✓ Sử dụng JavaScript để gửi và nhận dữ liệu giữa client và server.
- ✓ AJAX dựa trên:
 - JavaScript
 - XML
 - HTML
 - CSS
- ✓ AJAX sử dụng XML và yêu cầu HTTP



AJAX – CÁC BƯỚC

- ✓ Tạo Form xử lý HTML
- ✓ Thêm sự kiện xử lý
- ✓ Thêm đối tượng XMLHttpRequest
- ✓ Xử lý dữ liệu
- ✓ Lập trình xử lý phía server



Bước 1 – Tạo Form xử lý HTML

```
<html>
```

```
  <body>
```

```
    <form>
```

```
      Tên sản phẩm:
```

```
      <input type="text" id="txt1" />
```

```
    </form>
```

```
    <p>Danh sách tên sản phẩm:</p>
```

```
    <span id="txtHint"></span>
```

```
  </body>
```

```
</html>
```



Bước 2 - Thêm sự kiện xử lý

✓ Trong file script.js:

```
function showHint(str)
```

✓ Trong file ajax.php

```
<html>
```

```
  <body>
```

```
    <form>
```

Tên sản phẩm:

```
    <input type="text" onkeyup="showHint(this.value) "  
      id="txt1" >
```

```
  </form>
```

```
  <p>Danh sách tên sản phẩm:</p>
```

```
  <span id="txtHint"></span>
```

```
</body>
```



Đối tượng XMLHttpRequest

✓ Các phương thức

- **open()** : Thiết lập yêu cầu đến server (địa chỉ trang cần kết nối đến)
- **send()** : Gửi yêu cầu đến server.
- **abort()** : Hủy yêu cầu hiện tại

✓ Các thuộc tính

- **readyState**: Trạng thái hiện tại của đối tượng.
- **onreadystatechange**: Địa chỉ hàm gọi lại (callback)
- **responseText**: Chuỗi dữ liệu trả về.



Phương thức open

✓ void open(
 in DOMString method,
 in DOMString url);

✓ void open(
 in DOMString method,
 in DOMString url,
 in boolean async);

✓ void open(
 in DOMString method,
 in DOMString url,
 in boolean async,
 in DOMString user);



Phương thức open (tt)

```
✓ void open (  
    in DOMString method,  
    in DOMString url,  
    in boolean async,  
    in DOMString user,  
    in DOMString password) ;
```



Thuộc tính readyState

- ✓ **readyState=0**: Chưa khởi tạo: sau khi tạo đối tượng XMLHttpRequest nhưng chưa gọi open().
- ✓ **readyState=1**: Vừa khởi tạo: sau khi gọi open() nhưng chưa gọi send().
- ✓ **readyState=2**: Vừa gửi đi: ngay sau khi gọi send().
- ✓ **readyState=3**: Đang xử lý: sau khi kết nối đến server nhưng server chưa trả lời.
- ✓ **readyState=4**: Xong: sau khi server tính toán xong, dữ liệu đã gửi về xong.



Bước 3 – Thêm đối tượng XMLHttpRequest

```
✓ var xmlhttp = getHTTPObject();  
✓ function getHTTPObject() {  
    var xmlhttp;  
    if (window.XMLHttpRequest) {  
        /*Lấy đối tượng XMLHttpRequest đối với các browser  
        Firefox, Opera 8.0+, Safari, BlueBird,...*/  
        xmlhttp = new XMLHttpRequest();  
    } else if (window.ActiveXObject) {  
        //Lấy đối tượng ActiveXObject đối với browser IE  
        xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");  
    }  
    return xmlhttp;  
}
```



Bước 4 – Xử lý dữ liệu

//Thay đổi trạng thái của đối tượng dùng để hiển thị CSDL

```
function stateChanged()
```

```
{
```

```
    if (xmlHttp.readyState==4)
```

```
    {
```

```
        document.getElementById("txtHint").innerHTML=
```

```
        xmlHttp.responseText
```

```
    }
```

```
}
```



Bước 4 – Xử lý dữ liệu

//Hiển thị dữ liệu từ trang xử lý, gửi dữ liệu đến trang xử lý

```
function showHint(str){  
    if (str.length==0){  
        document.getElementById("txtHint").innerHTML="";  
        return}  
    xmlhttp = getHTTPObject()  
    if (xmlhttp==null){  
        alert ("Trình duyệt của bạn không hỗ trợ AJAX")  
        return}  
    var url="search.php"  
    url=url+"?q="+str  
    xmlhttp.onreadystatechange=stateChanged  
    xmlhttp.open("GET",url,true)  
    xmlhttp.send(null)}
```



Bước 5 - Lập trình xử lý phía server

✓ Hiển thị kết quả sẽ trả về (trong file getnameajax.php)

Đây là một trang tìm kiếm với biến đầu vào (Biến GET) là q.

