



jQuery & AJAX

Instructor: <Name of Instructor>

Latest updated by: HanhTT1

What is jQuery?

- jQuery is a library of JavaScript Functions.
- jQuery is a lightweight "write less, do more" JavaScript library.

What is jQuery?

- **The jQuery library contains the following features:**

- HTML element selections
- HTML element manipulation
- CSS manipulation
- HTML event functions
- JavaScript Effects and animations
- HTML DOM traversal and modification
- AJAX
- Utilities

Adding the jQuery Library to Your Pages

- The jQuery library is stored as a single JavaScript file, containing all the jQuery methods.
 - It can be added to a web page with the following mark-up:

```
<head>
  <script type="text/javascript" src = "jquery.js"></script>
</head>
```

Or you can use the hosted jQuery library from Google or Microsoft.

```
<head>
  <script type="text/javascript"
    src="http://ajax.googleapis.com/ajax/libs/jquery/1.6.2/jquery
    .min.js"></script>
</head>
```

© FPT Software

4

Basic jQuery Example

```
<!DOCTYPE html>
<html>
<head>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript">
$(document).ready(function(){
    $("button").click(function(){
        $("p").hide();
    });
});
</script>
</head>
<body>
<h2>This is a heading</h2>
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
<button>Click me</button>
</body>
</html>
```

© FPT Software

5

jQuery Syntax

- The jQuery syntax is tailor made for **selecting** HTML elements and perform some **action** on the element(s).
- Basic syntax is: **`$(selector).action()`**
 - A dollar sign to define jQuery
 - A (selector) to "query (or find)" HTML elements
 - A jQuery action() to be performed on the element(s)

jQuery Syntax- Examples

- `$(this).hide()` : hides current element.
- `$("p").hide()` : hides all paragraphs.
- `$("p.test").hide()` : hides all paragraphs with class="test".
- `$("#test").hide()` : hides the element with id="test"

jQuery Selectors - Examples

- `$('*')` : selects all elements.
- `$("p")` : selects all `<p>` elements.
- `$("p.intro")` : selects all `<p>` elements with `class="intro"`.
- `$("p#intro")` - selects the first `<p>` elements with `id="intro"`.
- `$(":animated")` : selects all elements that are currently animated.
- `$(":button")` : selects all `<button>` elements and `<input>` elements of `type="button"`.
- `$(":even")` : selects even elements.
- `$(":odd")` : selects odd elements.

jQuery Events

- The jQuery event handling methods are core functions in jQuery.
- Event handlers are methods that are called when "something happens" in HTML. The term "**triggered (or "fired") by an event**" is often used.
- It is common to put jQuery code into event handler methods in the <head> section

jQuery Events - Example

```
<!DOCTYPE html>
<html>
<head>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript">
$(document).ready(function(){
    $("button").click(function(){
        $("p").hide();
    });
});
</script>
</head>
<body>
<h2>This is a heading</h2>
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
<button>Click me</button>
</body>
</html>
```

© PPT Software

10

jQuery Events – Example (cont)

In the example above, a function is called when the click event for the button is triggered:

```
$("button").click(function() {..some code... })
```

The method hides all <p> elements:

```
$("p").hide();
```

jQuery Callback Functions

- A callback function is executed after the current animation is 100% finished.
- JavaScript statements are executed line by line. However, with animations, the next line of code can be run even though the animation is not finished. This can create errors.
- To prevent this, you can create a callback function.
- A callback function is executed after the current animation (effect) is finished

jQuery Callback Functions

- A callback function is executed after the current animation is 100% finished.
- JavaScript statements are executed line by line. However, with animations, the next line of code can be run even though the animation is not finished. This can create errors.
- To prevent this, you can create a callback function.
- A callback function is executed after the current animation (effect) is finished

jQuery Callback - Example

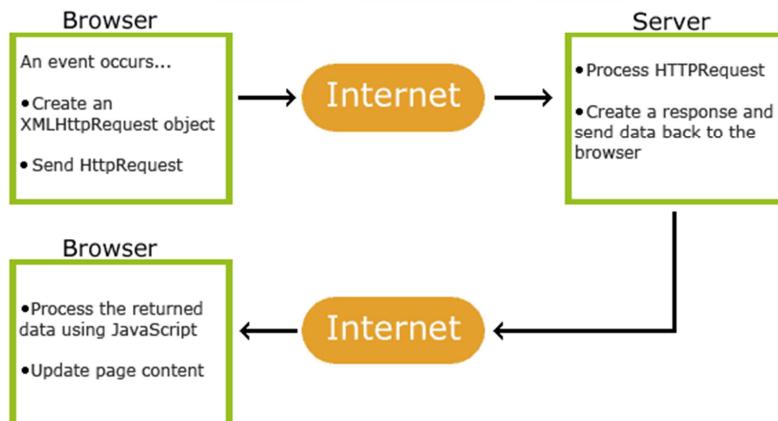
```
<!DOCTYPE html>
<html>
    <head>
        <script type="text/javascript" src="jquery.js"></script>
        <script type="text/javascript">
            $(document).ready(function(){
                $("button").click(function(){
                    $("p").hide(1000,function(){
                        alert("The paragraph is now hidden");
                    });
                });
            });
        </script></head>
        <body>
            <button>Hide</button>
            <p>This is a paragraph with little content.</p>
        </body>
    </html>
```

4

What is AJAX?

- AJAX = Asynchronous JavaScript and XML.
- AJAX is a technique for creating fast and dynamic web pages.
- AJAX allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page.
- Classic web pages, (which do not use AJAX) must reload the entire page if the content should change.
- Examples of applications using AJAX: Google Maps, Gmail, Youtube, and Facebook tabs.

How Ajax work?



AJAX is Based on Internet Standards

AJAX is based on internet standards, and uses a combination of:

- XMLHttpRequest object (to exchange data asynchronously with a server)
- JavaScript/DOM (to display/interact with the information)
- CSS (to style the data)
- XML (often used as the format for transferring data)

AJAX applications are browser- and platform-independent!

```
<!DOCTYPE html>
<html>
  <head>
    <script type="text/javascript">
      function loadXMLDoc()
      {
        var xmlhttp;
        if (window.XMLHttpRequest)
          {// code for IE7+, Firefox, Chrome, Opera, Safari
            xmlhttp=new XMLHttpRequest();
          }
        else
          {// code for IE6, IE5
            xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
          }
        xmlhttp.onreadystatechange=function()
        {
          if (xmlhttp.readyState==4 && xmlhttp.status==200)
            {
              document.getElementById("myDiv").innerHTML=xmlhttp.responseText;
            }
        }
        xmlhttp.open("GET" "ajax_info.txt",true);
        xmlhttp.send();
      }
    </script>
  </head>
  <body>
    <div id="myDiv"><h2>Let AJAX change this text</h2></div>
    <button type="button" onclick="loadXMLDoc()">Change Content</button>
  </body>
</html>
```

18

AJAX and jQuery

- jQuery provides a rich set of methods for AJAX web development.
- With jQuery AJAX, you can request TXT, HTML, XML or JSON data from a remote server using both HTTP Get and HTTP Post
- **And you can load remote data directly into selected HTML elements of your web page!**

Write Less, Do More

- The jQuery load() method is a simple (but very powerful) AJAX function. It has the following syntax:

`$(selector).load(url,data,callback)`

- Use the **selector** to define the HTML element(s) to change, and the **url parameter** to specify a web address for your data.

Only if you want to send data to the server, you need to use the *data* parameter. Only if you need to trigger a function after completion, you will use the *callback* parameter.

AJAX and jQuery - syntax

```
jQuery.ajax({  
    type:"POST",//Phuong thức gửi request là POST hoặc GET  
    data:"id=12&name=abc",//tham số gửi kèm  
    dataType:"xml",//kiểu dữ liệu trả về, mặc định là text  
    url:"/login/servletLogin",//Đường dẫn tới nơi xử lý  
    request ajax  
    success: function (){//hàm gọi về khi thực hiện thành  
    công  
        // mã lệnh  
    }  
});
```

AJAX and jQuery - Example

```
□  <html>
  <head>
    <script src="http://code.jquery.com/jquery-1.4.2.js"></script>
    <script type="text/javascript">
      jQuery(document).ready(function(){
        jQuery("#ajaxButton").click(function(){
          jQuery.ajax({
            type:"POST",
            url:"ajax.html",
            success:function(html){
              jQuery("#responseDiv").html(html);
            }
          });
        });
      });
    </script>
  </head>
  <body>
    <div>
      <input type="button" value="ajax" id="ajaxButton"/>
    </div>
    <div id="responseDiv"></div>
  </body>
</html>
```

2

AJAX and jQuery - Example

Tiếp tục tạo file **ajax.html** có nội dung như sau

```
<style>
    #sampleTable{
        border-collapse:collapse;
    }
    #sampleTable td{
        border: 1px solid black;
        width:100px;
    }
</style>
<table id="sampleTable">
    <tr>
        <td>Name</td>
        <td>Year</td>
    </tr>
    <tr>
        <td>Van A</td>
        <td>1982</td>
    </tr>
</table>
```

© 2011 Software

-3

AJAX and jQuery - Example

Mở file index.html bằng trình duyệt bất kỳ click vào button ajax bạn sẽ thấy toàn bộ nội dung của file ajax.html được lấy về trang index.html

- Khác nhau cơ bản giữa ajax thuần javascript và ajax với jquery (chủ yếu là cú pháp của jquery ngắn gọn hơn)



© FPT Software

25