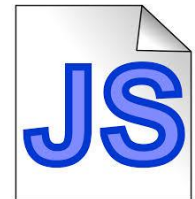


jQuery

Instructor:

Eng. Lama Barrak & Eng. Reem Naser



What is jQuery?

- jQuery is a lightweight, "write less, do more", JavaScript library.
- The purpose of jQuery is to make it much easier to use JavaScript on your website.
- jQuery takes a lot of common tasks that requires many lines of JavaScript code to accomplish, and wraps it into methods that you can call with a single line of code.
- jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

Adding jQuery to Your Web Pages

- To use jQuery, you need to download the jQuery library , and include it on the pages you wish to use it.
- The jQuery library is a single JavaScript file, and you reference to it using the HTML `<script>` tag:

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

- If you don't want to download and host jQuery yourself, you can include it from a CDN (Content Delivery Network).
- Both Google and Microsoft host jQuery.

Google CDN:

```
<head>  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>  
</head>
```

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 \$ sign to define/access jQuery
 - ✓ A (*selector*) to "query (or find)" HTML elements
 - ✓ A jQuery *action()* to be performed on the element(s)
- all jQuery methods in our examples, are inside a document ready event:

```
$(document).ready(function() {  
    // jQuery methods go here...  
});
```

- This is to prevent any jQuery code from running before the document is finished loading (is ready).

jQuery Syntax

- Example:

```
1 <html>
2 <head>
3 <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
4 <script>
5 $(document).ready(function() {
6     $("p").click(function() {
7         $(this).hide();
8     });
9 });
10 </script>
11 </head>
12 <body>
13
14 <p>If you click on me, I will disappear.</p>
15 <p>Click me away!</p>
16 <p>Click me too!</p>
17
18 </body>
19 </html>
```

jQuery Selectors

- jQuery selectors allow you to select and manipulate HTML element(s).
- With jQuery selectors you can find elements based on their id, classes, types, attributes, values of attributes and much more. It's based on the existing CSS Selectors, and in addition, it has some own custom selectors.
- All type of selectors in jQuery, start with the dollar sign and parentheses: \$().

Examples of jQuery Selectors

Syntax	Description
<code>\$("*")</code>	Selects all elements
<code>\$(this)</code>	Selects the current HTML element
<code>\$("p.intro")</code>	Selects all <code><p></code> elements with <code>class="intro"</code>
<code>\$("p:first")</code>	Selects the first <code><p></code> element
<code>\$("ul li:first")</code>	Selects the first <code></code> element of the first <code></code>
<code>\$("ul li:first-child")</code>	Selects the first <code></code> element of every <code></code>
<code>\$("[href]")</code>	Selects all elements with an <code>href</code> attribute
<code>\$("a[target='_blank']")</code>	Selects all <code><a></code> elements with a <code>target</code> attribute value equal to <code>"_blank"</code>
<code>\$("a[target!='_blank']")</code>	Selects all <code><a></code> elements with a <code>target</code> attribute value NOT equal to <code>"_blank"</code>
<code>\$(":button")</code>	Selects all <code><button></code> elements and <code><input></code> elements of <code>type="button"</code>
<code>\$("tr:even")</code>	Selects all even <code><tr></code> elements
<code>\$("tr:odd")</code>	Selects all odd <code><tr></code> elements

Examples of jQuery Selectors

Syntax	Description
<code>\$("p#intro:first")</code>	Selects the first <code><p></code> element with <code>id="intro"</code>
<code>\$(".intro")</code>	Selects all elements with <code>class="intro"</code>
<code>\$("#intro")</code>	Selects the first element with <code>id="intro"</code>
<code>\$("[href\$='.jpg']")</code>	Selects all elements with an <code>href</code> attribute that ends with <code>".jpg"</code>
<code>\$("div#intro .head")</code>	Selects all elements with <code>class="head"</code> inside a <code><div></code> element with <code>id="intro"</code>

jQuery Selectors

- Example :

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
5 <script>
6 $(document).ready(function() {
7 //
8 $("button").click(function() {
9 //
10     $(this).hide();
11     //$("#p").hide();
12 //$("#p#p1").hide();
13 //$("#p.c2").hide();
14 //$("#*").hide();
15 });
16 });
17 </script>
18 </head>
19 <body>
20 <h1> My Application </h1>
21 <p>paragraph </p>
22 <p id="p1" class="c1">first paragraph</p>
23 <p id="p2" class="c2">second paragraph</p>
24 <button>Click me</button>
25
26 </body>
27 </html>
```

jQuery Event Methods

Event Method	Description
<code>\$(document).ready(function)</code>	Binds a function to the ready event of a document (when the document is finished loading)
<code>\$(selector).click(function)</code>	Triggers, or binds a function to the click event of selected elements
<code>\$(selector).dblclick(function)</code>	Triggers, or binds a function to the double click event of selected elements
<code>\$(selector).focus(function)</code>	Triggers, or binds a function to the focus event of selected elements
<code>\$(selector).mouseover(function)</code>	Triggers, or binds a function to the mouseover event of selected elements

jQuery Event Methods

- Example :

```
1 <html>
2 <head>
3 <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
4 <script>
5 $(document).ready(function(){
6     $("p").dblclick(function(){
7         $(this).hide();
8     });
9 });
10 </script>
11 </head>
12 <body>
13
14 <p>If you double-click on me, I will disappear.</p>
15 <p>Click me away!</p>
16 <p>Click me too!</p>
17
18 </body>
19 </html>
```

jQuery Effects- Hide and Show

- Example :

```
<head>
<script src="jquery.js"></script>
<script>
$(document).ready(function() {
    $("#hide").click(function() {
        $("p").hide();
    });
    $("#show").click(function() {
        $("p").show();
    });
});
</script>
</head>
<body>
<p>If you click on the "Hide" button, I will disappear.</p>
<button id="hide">Hide</button>
<button id="show">Show</button>
</body>
```

jQuery Effects- Hide and Show

Syntax:

```
$(selector).hide(speed,callback);  
$(selector).show(speed,callback);
```

The optional speed parameter specifies the speed of the hiding/showing, and can take the following values: "slow", "fast", or milliseconds.

The optional callback parameter is the name of a function to be executed after hide (or show) completes.

- **jQuery toggle()**

With jQuery, you can toggle between the hide() and show() methods with the toggle() method.

Shown elements are hidden and hidden elements are shown:

Example

```
$("#button").click(function() {  
    $("#p").toggle();  
});
```

jQuery Effects- Fading

- With jQuery you can fade an element in and out of visibility.

Method	Description
<u>fadeIn()</u>	Fades in the selected elements
<u>fadeOut()</u>	Fades out the selected elements
<u>fadeTo()</u>	Fades in/out the selected elements to a given opacity
<u>fadeToggle()</u>	Toggles between the <u>fadeIn()</u> and <u>fadeOut()</u> methods

jQuery Effects- Fading

```
1 <head>
2 <script src="jquery.js"></script>
3 <script>
4 $(document).ready(function(){
5     $("button").click(function(){
6         $("#div1").fadeIn();
7         $("#div2").fadeIn("slow");
8         $("#div3").fadeIn(3000);
9     });
10 });
11 </script>
12 </head>
13
14 <body>
15 <p>Demonstrate fadeIn() with different parameters.</p>
16 <button>Click to fade in boxes</button>
17 <br><br>
18 <div id="div1" style="width:80px;height:80px;display:none;background-color:red;"></div><br>
19 <div id="div2" style="width:80px;height:80px;display:none;background-color:green;"></div><br>
20 <div id="div3" style="width:80px;height:80px;display:none;background-color:blue;"></div>
21
22 </body>
```

jQuery Effects- Sliding

- The jQuery slide methods slides elements up and down.

Method	Description
<u>slideDown()</u>	Slides-down (shows) the selected elements
<u>slideToggle()</u>	Toggles between the <u>slideUp()</u> and <u>slideDown()</u> methods
<u>slideUp()</u>	Slides-up (hides) the selected elements

- Example :

```
1 <head>
2 <script src="jquery.js"></script>
3 <script>
4 $(document).ready(function() {
5     $("#flip").click(function() {
6         $("#panel").slideDown("slow");
7     });
8 });
9 </script>
```


jQuery Effects- Sliding

- Example Con. :

```
10
11 <style type="text/css">
12 #panel,#flip
13 {
14 padding:5px;
15 text-align:center;
16 background-color:#e5eccc;
17 border:solid 1px #c3c3c3;
18 }
19 #panel
20 {
21 padding:50px;
22 display:none;
23 }
24 </style>
25 </head>
26 <body>
27
28 <div id="flip">Click to slide down panel</div>
29 <div id="panel">Hello world!</div>
30
31 </body>
```

jQuery Effects - Animation

- The jQuery animate() method lets you create custom animations.

Syntax:

```
$(selector).animate({params}, speed, callback);
```

The required params parameter defines the CSS properties to be animated.

The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.

The optional callback parameter is the name of a function to be executed after the animation completes.

The following example demonstrates a simple use of the animate() method; it moves a <div> element to the left, until it has reached a left property of 250px:

Example

```
$("button").click(function() {  
    $("div").animate({left:'250px'});  
});
```

jQuery Effects - Animation

```
1 <head>
2 <script src="jquery.js"></script>
3 <script>
4 $(document).ready(function() {
5     $("button").click(function() {
6         $("div").animate({left:'250px'});
7         // $("div").animate({height:300}, "slow");
8         // $("div").animate({width:"70%", opacity:0.4, marginLeft:"0.6in", fontSize:"3em"});
9     });
10 });
11 </script>
12 </head>
13
14 <body>
15 <button>Start Animation</button>
16 <p>jQuery animation</p>
17 <div style="background:#98bf21;height:100px;width:100px;position:absolute;">
18 </div>
19
20 </body>
21
```

jQuery Callback Functions

- A callback function is executed after the current effect is 100% finished.
- Typical syntax: **`$(selector).hide(speed,callback);`**
- Example :

Example with Callback

```
$("#button").click(function() {  
    $("#p").hide("slow",function() {  
        alert("The paragraph is now hidden");  
    });  
});
```

jQuery - Add Elements

- With jQuery, it is easy to add new elements/content.
 - `append()` - Inserts content at the end of the selected elements
 - `prepend()` - Inserts content at the beginning of the selected elements
 - `after()` - Inserts content after the selected elements
 - `before()` - Inserts content before the selected elements
-

- Example :

```
$("p").append("Some appended text.");
```

- `html()` - Sets or returns the content of selected elements (including HTML markup)
 - Example : `$("#test1").html("Hello world!");`

jQuery css() Method

- The `css()` method sets or returns one or more style properties for the selected elements.
- To return the value of a specified CSS property, use the following syntax:
 - `css("propertyname");`
- To set a specified CSS property, use the following syntax:
 - `css("propertyname","value");`
- To set multiple CSS properties, use the following syntax:
 - `css({"propertyname":"value","propertyname":"value",...});`

jQuery css() Method

- Example:

```
<script>
$(document).ready(function() {

    $("button#b1").click(function() {
        $("#div1").css("border", "5px red dotted");
    });

    $("button#b2").click(function() {
        $("#div2").css({ "background-color": "yellow", "border": "4px blue dashed" } );
    });

    $("button#b3").click(function() {
        $("#div3").css("border", $("#div1").css("border"));
    });

});
</script>
```

jQuery-Chaining

- Chaining allows us to run multiple jQuery methods (on the same element) within a single statement.
- This way, browsers do not have to find the same element(s) more than once.
- To chain an action, you simply append the action to the previous action.
- The following example chains together the `css()`, `slideUp()`, and `slideDown()` methods. The "p1" element first changes to red, then it slides up, and then it slides down:
 - `$("#p1").css("color", "red").slideUp(2000).slideDown(2000);`

Questions

