INTRODUCING JQUERY - FILTERS AND SELECTORS

Unit 4

INDEX

1 – INTRODUCING JQUERY

- “There are only two kind of languages: the ones people complain about and the ones nobody uses”

  Bjarne Stroustrup – C++ Designer

- Ajaxm Prototype, Node.js, jQuery, etc…
- jQuery is used by 73% of the top million websites.
- Is a JS library created in 2006 designed to simplify the client-side scripting.
- https://api.jquery.com/

1 – INTRODUCING JQUERY

- Write less, do more.

```
var checkedValue =
  jQuery('input:radio[name="some-radio-group"]:checked').val();
```

2 – DOWNLOADING JQUERY

- Download jQuery from its official website.
- Save it in your local machine.
- Link it from your webpage.

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>jQuery Hello World DWEC</title>

  <script type="text/javascript" src="js/jquery-3.1.1.min.js"></script>

</head>
<body>

</body>
</html>
```

3 – HELLO WORLD

- JavaScript

```
window.onLoad = function(event) {

    alert("¡La página acaba de cargar!");

}
```

- The event is triggered when all the content of the page have been loaded (including images).

3 – HELLO WORLD

- jQuery

```
$("document").ready(function () {

    alert("Web page is loaded!!!");

});
```

- The event is triggered when the DOM has been downloaded, not all the content of the webpage.

4 – SELECTING ELEMENTS

- Selectors allow us to get content from the document to manipulate it.
- It return a jQuery object with multiple funcitons and properties to interact with.
- We have several selectors in jQuery:
  - Simples
  - Composite
  - We can also use filters
- http://api.jquery.com/category/selectors/

4 – SELECTING ELEMENTS

- Basic selectors are based on the CSS syntax and work in the same way so far.
- We can check the selection with .length

```
if ($('div.foo').length) { ... }
```

Selector	Description	Example
tag	Get the elements by the html tag	$("div");
#id	Get the elements by the id	$("myimg");
.class	Get the elements by the class name	$(".myclass");
tag.class	Get the elements of type tag with the class specified	$("ul.customclass");
tag#id.class	Get the elements of type tag with the specified id and class	$("form#input1.myclass");
*	Get all the elements of the page	$("*");

4 – SELECTING ELEMENTS

- Composite selectors allow get objects by its hierarchy and combination.
- Some of them are:

Selector	Description	Example
selector, selector, selector...	Get all the specified selectors	$("p, ul.b");
.class1.class2	Get the elements with class1 and class2	$(".a.b");
father>child...	Get all the direct childs of the father	$("ul.customclass>li>a");

Selector	Description	In CSS?
E F	Matches all elements with tag name F that are descendants of E	✓
E>F	Matches all elements with tag name F that are direct children of E	✓
E+F	Matches all elements with tag name F that are immediately preceded by sibling E	✓
E~F	Matches all elements with tag name F preceded by any sibling E	✓

5 – BASIC FILTERS

- Filters keep the simplicity of selecting elements in jQuery and are used to polish the results of a selector
- There are many types of filters, now, we are going to study the basic ones.

Filter	Description	Example
:first	Get the first element	$("div:first");
:last	Get the last element	$("div:last");
:even // :odd	Get the pair/impair elements	$("div:even");//$("div:odd");
:eq(n)//:gt(n)//:lt(n)	Get the elements equal/greater or lower than the specified index	$("div:lt(4)");
:animated	Get the elements being animated	$("div:animated");
:not(selector)	All the elements but the ones that meet with the selector provided	$("div:not(div:eq(2)");

6 – ADVANCED FILTERS

- Attribute filters
  - They allow us to refine the results gathered by the selector using the attributes of the element.
  - Attribute selectors are extremely powerful and allow you to select elements based on their attributes.
  - You can easily recognize these selectors because they"re wrapped with square brackets (for example, [selector]).
  - They can be very slow.

6 – ADVANCED FILTERS

- Attribute filters
  - We can have multuple filters working as an AND. [filter][filter]

Filter	Description	Example
[attributeName]	Get element that contain a specified attribute	$(“form[method]”);
[attributeName=value]	Get the element with the given attribute and with the given value. You can also use !=.	$(“div[id=‘container1’]);
[attributeName^=value]	Get the element with the given attribute and with the value beginning with the given value. You can also use !^	$("div[id^=‘container’]");
[attributeName$=value]	Get the element with the given attribute and with the value finishing with the given value. You can also use !$	$("a[href^=‘.pdf’]");
[attributeName*=value]	Get the element with the given attribute and with the value containing the given value. You can also use !*	$("a[href*=‘jquery.com’]");

6 – ADVANCED FILTERS

- Content filters
  - Allow us to refine the results gathered by the selector using the content of the element.

Filter	Description	Example
:contains(text)	Get elements that contains the specified text	$(“div:contains(‘my house’)”);
:empty	Get empty elements.	$(“div:empty”);
:has(selector)	Get elements that contains a element with the specified selector.	$("div:has(p[class=a])");
:parent	Get the parent elements (containing at least one element)	$(“div:parent”);

6 – ADVANCED FILTERS

- Visibility filters
  - Allow us to refine the results gathered by the selector depending if the elements are visible or not.

Filter	Description	Example
:visible	Get the visible elements	$("div:visible");
:hidden	Get the hidden elements	$("div:hidden");

6 – ADVANCED FILTERS

- Child filters
  - Allow us to refine the results gathered by the selector considering its relationship with their parents.

Filter	Description	Example
:nth-child(index)	The element at the specified index	$("div p:nth-child(2)");
:nth-child(even) // :nth-child(odd)	Even/odd elements	$("div p:nth-child(even)"); // $("div p:nth-child(odd)");
:first-child // :last-child	Get first/last child of a element	$("div p:first-child"); // $("div p:last-child");
:only-child	Get the child without siblings	$("div p:only-child");

6 – ADVANCED FILTERS

- Form filters
  - Very similar to the others but useful to find specific element in a form.

Selector	Description	In CSS?
`:checkbox`	Selects only check box elements (`input[type=checkbox]`)	
`:checked`	Selects check boxes or radio elements in the checked state or `options` of `select` elements that are in a selected state	✓
`:disabled`	Selects only elements in the disabled state	✓
`:enabled`	Selects only elements in the enabled state	✓
`:file`	Selects only file input elements (`input[type=file]`)	
`:focus`	Selects elements that have the focus at the time the selector is run	✓
`:image`	Selects only image input elements (`input[type=image]`)	
`:input`	Selects only form elements (`input`, `select`, `textarea`, `button`)	
`:password`	Selects only password elements (`input[type=password]`)	
`:radio`	Selects only radio elements (`input[type=radio]`)	
`:reset`	Selects only reset buttons (`input[type=reset]` or `button[type=reset]`)	
`:selected`	Selects only `option` elements that are in the selected state	
`:submit`	Selects only submit buttons (`button[type=submit]` or `input[type=submit]`)	
`:text`	Selects only text elements (`input[type=text]`) or `input` without a `type` specified (because `type=text` is the default)	