

DOM:

- The document object model is an cross-platform, language independent application programming Interface (API) that defines a standard way for programs to access HTML document
- Originally developed by Netscape, then Microsoft developed its own version
- Represents a HTML document as a tree of nodes in which each node represents an element In HTML document
- Document object is the root of DOM tree

JQuery: (replaces the DOM javascript commands so <table> but in DOM etc)

- Client-side JavaScript library released in 2006 by John Resig
- JQuery code is a piece of JavaScript code.
- It takes common task that require many lines of JavaScript code and wraps them into methods that you can call with a single line of code

JQuery Functions:

- HTML and DOM navigation and manipulation
- CSS manipulation
- Event registration and handling (events)
- Developing with AJAX
- Other utilities

Format for JQuery BareBone:

- What you put before any jQuery events take place
- Safer in that it ensure that you only start manipulating DOM after HTML is fully loaded.
- Contains: JQuery methods + Event handler
- Only one of these to store everything

```
----- jQuery BareBone -----  
<script>  
  $(document).ready(function() {  
    // Other jQuery code goes here  
  });  
</script>  
-----
```

Format for JQuery Events triggers → Methods:

- Similar to JavaScript events-
 - These events execute (usually a script in-built) when something has occurred to the HTML element/tag
- Minimises the amount of coding required if you used JavaScript event handler.
- Required since it indicates when the JQUERY *method* will execute

```
$("#[selector]").[EventAttribute](function() {  
    /*Query Methods*  
});
```

Format for JQuery Methods: (Inside javascript BareBone + Inside Event handler):

```
$("#[selector]").method().method().method()
```

- \$ ← stands for jQuery
- [selector] doesn't include < > tags
 - [selector]=\$(this)
 - References the selector above. But refers to the element that has triggered event attribute ie not all the selectors of the whole HTML just selected one
- Steps: (two things)
 - \$/jQuery function takes a parameter [selector]

- The function will analyse the parameter and decide how many of HTML element match selection criteria
- All the element matching will form an array
- Return JQuery/\$ object will have many action *methods*

Types of Selectors:

Basic:

- For one element in selector

such as element, #id, class, .classA, .classB, #id

```
$('.p'), $('#id'), $('.class'), $('.classA, classB'), $("h1,div,p")
```

Hierarchy:

```
Eg: $('form input'), $('#main > *'), $('label + input')
```

Form element selector:

Selectors	Matched Elements
:input	input, select, textarea and button elements
:text, :radio, :checkbox, :image, :submit, :reset, :password, :file	input element and attribute that is equal to the specified selectors
:button	button element, input element with type "button"

```
var allInputs = $( ":input" )
```

Event Methods/*Attributes* (function) of JQuery Object:

Mouse Events	Keyboard Events	Form Events	Document/Window Events
click	keypress	submit	load
dblclick	keydown	change	resize
mouseenter	keyup	focus	scroll
mouseleave		blur	unload

28

Hover: Triggers two function when mouse hovers and then leaves

Window.load: so when window loads methods will occur

Method (functions) of JQuery Object:

- These are executable (functions) they do things to JQuery *object*
- **Visual effects methods:**
 - Basic visual effects, eg
 - show(), show(speed)
 - hide(), hide(speed), hide(speed, callback)
 - toggle(), toggle(speed)
 - Fading, eg
 - fadeIn(speed),
 - fadeOut(speed),
 - fadeTo(speed, opacity)
 - Slide
 - slideDown(), slide(speed, callback)
 - slideUp(), slideUp(speed, callback)
 - slideToggle(), slideToggle(speed, callback)
 - More, such as animate, stop etc

- Add new HTML Content:

Add Elements/Contents

- The following jQuery methods are used to add *new HTML 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

`remove()` - Removes the selected element AND its child elements

`empty()` - Removes the child elements FROM the selected element(s)

- Manipulate CSS:

Manipulate CSS

jQuery has several methods for CSS manipulation:

- `addClass()` - Adds one or more classes to the selected elements
- `removeClass()` - Removes one or more classes from the selected elements
- `toggleClass()` - Toggles between adding/removing classes from the selected elements
- `css()` - Sets or returns the style attribute
- Refer [w3schools](#) for examples

- Working with Dimension of element

- jQuery has several important methods for working with dimensions:
 - `width()` - sets or returns the width of an element
 - `height()` – sets/returns the height of an element
 - `innerWidth()`
 - `innerHeight()`
 - `outerWidth()`
 - `outerHeight()`
 - Refer w3schools for examples

- **HTML Document Traversing:**

- Are used to find HTML elements based on their relative position to other elements

Traversing Up

- Three useful jQuery methods for traversing up the DOM tree are:
 - `parent()` - returns the parent element of the selected element; i.e. a single step up the tree
 - `parents()` - returns all ancestor elements of the selected element; i.e. all the way up to the root element of the document tree
 - `parentsUntil()` - returns all ancestor elements between the selected element and a given argument

- Two useful jQuery methods for traversing down the DOM tree are:
 - `children()` - returns all direct children of each occurrence of the selected element; i.e., a single step down the tree
 - You can filter the search with an optional parameter
 - `find()` - returns all descendant elements of the selected element all the way down to the last descendant

- There are many useful jQuery methods for traversing sideways within the DOM tree:
 - `siblings()` - returns all sibling elements of the selected element
 - You can filter the search with an optional parameter
 - `next()`, `nextAll()`, `nextUntil()`
 - `prev()`, `prevAll()`, `prevUntil()`
- Refer w3schools for examples related to traversing using sibling methods

- Filtering methods:

Filter Methods

The most basic filtering methods are:

- `first()`: returns the first element of the specified elements
- `last()` : returns the last element of the specified elements
- `eq()` : returns an element with a specific index number of the selected elements
- The index number starts at 0

Properties of JQuery Object:

- These are properties (Values) associated with an JQuery *object*
- We can retrieve/modify these properties of an *object* and do something with them

- `text()` - returns the text content of all selected elements (HTML tags removed)
- `text(string)` – set the contents of all selected elements to text *string* (if there are HTML tags, they are treated as normal text)
- `html()` – return the content of the first of the selected elements (including HTML markup if there are)
- `html(string)` - sets the contents of selected elements to *string* (including HTML markup)

- `val()` - returns the value of the first of the selected form elements
- `val(string)` - sets the values of selected form elements to *string*
- `attr(attr-name)` – returns the value of the specified attribute in the first of the selected elements.
- `attr(attr-name, attr-value)` – set the value of the specified attribute *attr-name* in all selected elements to *attr-value*.
- The `attr` method also allows us to set several attributes at the same time:

```
attr( { attr-name1 : attr-val1,
      attr-name2 : attr-val2,
      . . . } )
```

.Val: For value of input