

## Session 01: Introduction to JQuery

### Topics

- Introduction of JQuery
- JQuery Anatomy
- JQuery Features
- How to use JQuery
- Selectors
- DOM Manipulation
- Events
- Ajax with JQuery

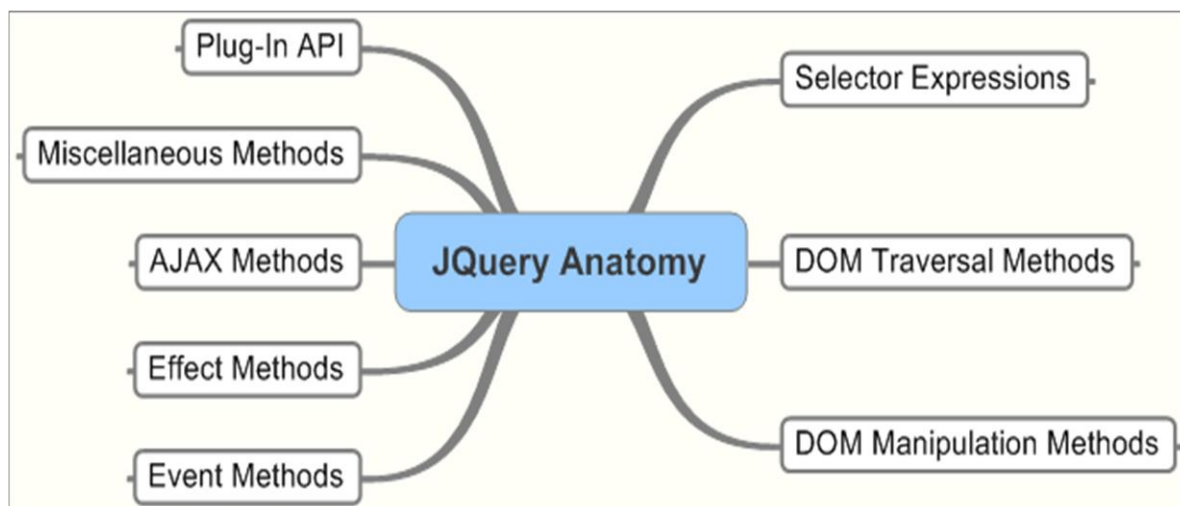
### Introduction of jQuery

jQuery is a fast and concise JavaScript Library created by John Resig in 2006 with a nice motto – Write less, do more.

jQuery simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development.

jQuery is a JavaScript toolkit designed to simplify various tasks by writing less code.

### Anatomy of JQuery



## JQuery Features

**DOM manipulation** – The jQuery made it easy to select DOM elements, traverse them and modifying their content by using cross-browser open source selector engine called **Sizzle**.

**Event handling** – The jQuery offers an elegant way to capture a wide variety of events, such as a user clicking on a link, without the need to clutter the HTML code itself with event handlers.

**AJAX Support** – The jQuery helps you a lot to develop a responsive and feature-rich site using AJAX technology.

**Animations** – The jQuery comes with plenty of built-in animation effects which you can use in your websites.

**Lightweight** – The jQuery is very lightweight library - about 19KB in size

( Minified and gzipped ).

**Cross Browser Support** – The jQuery has cross-browser support, and works well in IE 6.0+, FF 2.0+, Safari 3.0+, Chrome and Opera 9.0+

**Latest Technology** – The jQuery supports CSS3 selectors and basic XPath syntax.

## How to use JQuery

There are two ways to use jQuery.

- Local Installation – You can download jQuery library on your local machine and include it in your HTML code.

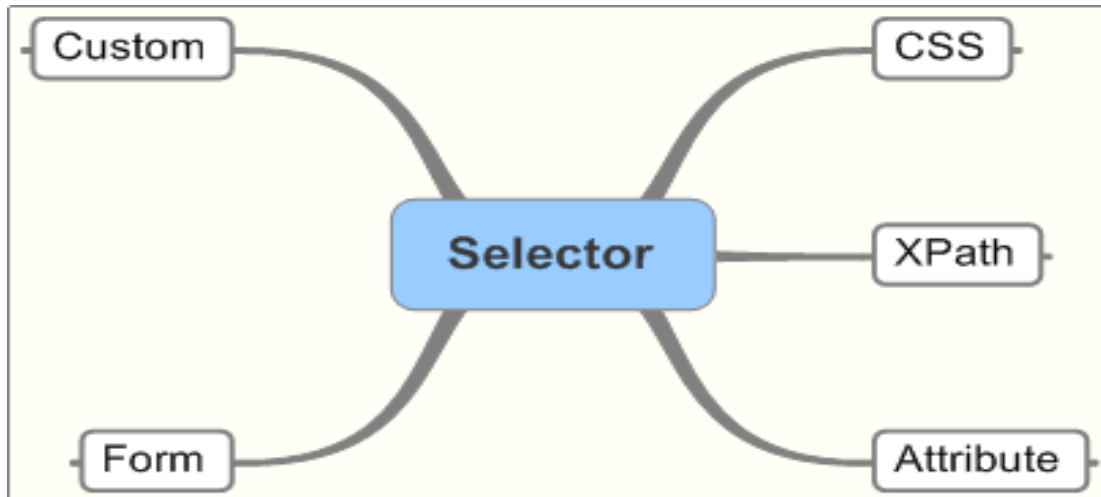
```
<script type="text/javascript" src="/jquery/jquery-1.11.3.min.js"></script>
```

- CDN Based Version – You can include jQuery library into your HTML code directly from Content Delivery Network (CDN).

```
<script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js">
</script>
```

## Selector

A jQuery Selector is a function which makes use of expressions to find out matching elements from a DOM based on the given criteria. Simply you can say, selectors are used to select one or more HTML elements using jQuery. Once an element is selected then we can perform various operations on that selected element.



## Basic Selectors

### Select Elements by ID

- Replacement of **document.getElementById** in JavaScript.
- Use # character to select elements by ID.

Example: `$('#myID')` – Selects an element with ID myID.

### 2. Select Elements by Class

- Selects all the elements with the specified class name.
- Use dot(.) character to select elements by class name.

Example: `$('.myClass')` -Selects an element with class 'myClass'.

### 3. Select Elements by Name

- Allows to find an element based on name.

Example:

`$("p")` - All <p> elements

### 4. Select Elements by Attribute

- Allows to find an element based on attributes set on it.
- Attribute name can be specified in square brackets  
e.g. `$('[class]')` - will return all the elements that have class attribute.

## Selectors

Selector	Example	Selects
*	<code>\$("*")</code>	All elements
#id	<code>\$("#lastname")</code>	The element with id="lastname"
.class	<code>\$(".intro")</code>	All elements with class="intro"
.class,.class	<code>\$(".intro,.demo")</code>	All elements with the class "intro" or "demo"
element	<code>\$("p")</code>	All <p> elements
el1,el2,el3	<code>\$("h1,div,p")</code>	All <h1>, <div> and <p> elements
:first	<code>\$("p:first")</code>	The first <p> element
:last	<code>\$("p:last")</code>	The last <p> element
:even	<code>\$("tr:even")</code>	All even <tr> elements
:odd	<code>\$("tr:odd")</code>	All odd <tr> elements
:first-child	<code>\$("p:first-child")</code>	All <p> elements that are the first child of their parent
:last-child	<code>\$("p:last-child")</code>	All <p> elements that are the last child of their parent
:nth-child(n)	<code>\$("p:nth-child(2)")</code>	All <p> elements that are the 2nd child of their parent

:only-child()	\$("p:only-child")	All <p> elements that are the only child of their parent
:empty	\$(":empty")	All elements that are empty
:visible	\$("table:visible")	All visible tables
:hidden	\$("p:hidden")	All hidden <p> elements
:has(selector)	\$("div:has(p)")	All <div> elements that have a <p> element
[attribute]	\$("[href]")	All elements with a href attribute
[attribute=value]	\$("[href='default.htm']")	All elements with a href attribute value equal to "default.htm"
[attribute!=value]	\$("[href!='default.htm']")	All elements with a href attribute value not equal to "default.htm"
[attribute^=value]	\$("[title^='Tom']")	All elements with a title attribute value starting with "Tom"
:input	\$(":input")	All input elements
:text	\$(":text")	All input elements with type="text"
:password	\$(":password")	All input elements with type="password"
:radio	\$(":radio")	All input elements with type="radio"
:checkbox	\$(":checkbox")	All input elements with type="checkbox"
:submit	\$(":submit")	All input elements with type="submit"
:button	\$(":button")	All input elements with type="button"
:image	\$(":image")	All input elements with type="image"
:selected	\$(":selected")	All selected input elements
:checked	\$(":checked")	All checked input elements

## JQuery Example

```
<!DOCTYPE html>
<html>
  <head>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js">
    </script>
    <script>
      $(document).ready(function() {
        $("span").hide();
      });
    </script>
    <style>
      .heading{ font-weight:bold;font-size:40px;background-color:grey;}
    </style>
  </head>
  <body>
    <div id="Assets">
      <p class="heading">Assets</p>
      <span>Details</span>
    </div>
  </body>
</html>
```

## Try JQuery

- Hide the Register section in the page.
- Highlight the Latest Books section.
- Highlight the headers with different background color.
- Change the background color of input tags to gray.
- 

## DOM Manipulation

Methods	Syntax	Description
addClass(), removeClass()	\$("#div").addClass("highlight") \$("#div").removeClass("highlight")	Adds/Removes the specified class(es) to each element in the set of matched elements.
toggleClass()	\$("#div").toggleClass("highlight")	Add or remove one or more classes
remove()	\$("#div1").remove()	Remove the specified element from DOM

append(), prepend()	<code>\$("p").append("text")</code> <code>\$("p").prepend("text.")</code>	Insert content to the end/Insert the content at the beginning
appendTo(), prependTo()		append/prepend
after(), before(),	<code>\$("img").after("text after")</code> <code>\$("img").before("text before")</code>	Insert content after/before each element in the set of matched elements.
empty()	<code>\$("#div1").empty()</code>	Remove all child nodes of the set of matched elements from the DOM.
Set Content - text(), html(), and val()	<code>\$("#test1").text("Hello world!")</code> <code>\$("#test2").html("&lt;b&gt;Hello world!&lt;/b&gt;")</code> <code>\$("#test3").val("Hello world")</code>	Sets or gets the content.

## Events

Events are actions that can be detected by your Web Application. Event methods trigger or attach a function to an event handler for the selected elements.

- Browser Events
- Document Loading
- Attaching Event Handlers
- Form Events
- Keyboard Events
- Mouse Events

Methods	Syntax	Description
resize()	<code>\$ ( window ).resize()</code>	Browser events
scroll()	<code>\$ ( window ).scroll()</code>	

load() unload() ready()	<code>\$( window ).load()</code> <code>\$( window ).unload()</code> <code>\$( document ).ready( handler )</code>	Document loading events
bind() unbind() on() off() trigger() delegate()	<code>\$( "#foo" ).bind( "click", function() {});</code> <code>\$( "#foo" ).unbind( "click" );</code> <code>\$( "table" ).on( "click", "td", function() {});</code> <code>\$( "p" ).off();</code> <code>\$( "#foo" ).trigger( "click" );</code> <code>\$( "table" ).delegate( "td", "click", function() {});</code>	Attaching event handlers
blur() change()	Attaches/Triggers the blur event Attaches/Triggers the change event	
keydown() keypress () keyup()	Attaches/Triggers the keydown event Attaches/Triggers the keypress event Attaches/Triggers the keyup event	
click() dblclick() mousedown() mouseenter () mouseleave() hover()	Attaches/Triggers the click event Attaches/Triggers the dblclick event Attaches/Triggers the mousedown event Attaches/Triggers the mouseenter event Attaches/Triggers the mouseleave event Attaches/Triggers the hover event	click() dblclick() mousedown() mouseenter () mouseleave() hover()

## AJAX with JQuery

JQuery provides several methods for AJAX functionality. With the jQuery AJAX methods, you can request text, HTML, XML, or JSON from a remote server using both HTTP Get and HTTP Post - And you can load the external data directly into the selected HTML elements of your web page.



Methods	Syntax	Description
ajax()	\$.ajax()	Load a remote page using an HTTP request
load()	\$.load()	Load HTML from a remote file and inject it into the DOM.
get()	\$.get()	Load a remote page using an HTTP GET request.
getJSON()	\$.getJSON()	Load JSON data using an HTTP GET request.
getScript()	\$.getScript()	Loads, and executes, a local JavaScript file using an HTTP GET request.
post()	\$.post()	Loads HTML by performing an HTTP post request.

## Effects

Methods	Description
Basics	hide(),show(),toggle()
Sliding	slideDown(),slideUp(),slideToggle()
Fading	fadeIn(),fadeOut(),fadeTo(),fadeToggle()
Custom	animate(),stop()

## Try JQuery

- Highlight the third book in the latest books section.
- Highlight the books on mouse click.
- Display a message to the user at the end of the page with all user input details on click of submit button.
- Cancel all the user inputs on click of Cancel button.
- Display the width of the page on Resizing at the end of the page.
- Read from a JSON data file and display at the end of the page.

## Reference

- [www.try.jquery.com](http://www.try.jquery.com)
- <http://www.tutorialspoint.com/jquery/>
- <https://learn.jquery.com>