

# JQuery



Rajeev Gupta  
Rgupta.mtech@gmail.com  
Java Trainer & Consultant

# jQuery



Currently, uses of jQuery is a most popular JavaScript library that works on all the browsers. It also supports CSS3 selectors and basic X Path syntax.

jQuery is a **JavaScript library** that allows **web** developers to add extra functionality to their websites.

4. Ease of learning
5. Write Less, Do More
6. Search engine optimized
7. DOM Manipulation
8. Easy integration with Visual Studio IDE
9. Even when JavaScript is disabled, a jQuery element has still displayed
10. jQuery makes flash like animated applications

# What is jQuery?

- jQuery is a lightweight, "write less, do more", JavaScript library.
- jQuery greatly simplifies JavaScript programming.
- The main purpose of jQuery is to make it much easier to use JavaScript on your website.
- jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.

- There are lots of other JavaScript frameworks out there, but jQuery seems to be the most popular, and also the most extendable.
- Many of the biggest companies on the Web use jQuery, such as:
  - Google
  - Microsoft
  - IBM
  - Netflix

# jQuery syntax

## About Syntax

jQuery alias



`$(selector).action()`

\$ is an identifier,  
which is used to  
access all  
functions

A **(selector)** is  
used to find  
HTML elements.

jQuery **.action()**  
to be performed  
on the elements.

# jQuery selectors

## SELECTORS

jQuery alias

`$(selector).action()`

- The element Selector
- The #id Selector
- The .class Selector

**What are  
id's and  
classes?**

# jQuery selectors

## THE #ID SELECTORS

- The jQuery *#id* selector uses the id attribute of an HTML tag to find the specific element.

`$("#idName").action()`

## THE .CLASS SELECTORS

- The jQuery *.class* selector finds elements with a specific class

`$(".className").action()`

# Some selectors

## SOME MORE SELECTORS

Syntax	Description
<code>\$("*")</code>	Selects all elements
<code>\$(this)</code>	Selects the current HTML element
<code>\$("p.intro")</code>	Selects all <code>&lt;p&gt;</code> elements with <code>class="intro"</code>
<code>\$("p:first")</code>	Selects the first <code>&lt;p&gt;</code> element
<code>\$("ul li:first")</code>	Selects the first <code>&lt;li&gt;</code> element of the first <code>&lt;ul&gt;</code>
<code>\$("ul li:first-child")</code>	Selects the first <code>&lt;li&gt;</code> element of every <code>&lt;ul&gt;</code>
<code>\$("[href]")</code>	Selects all elements with an <code>href</code> attribute
<code>\$("a[target='_blank']")</code>	Selects all <code>&lt;a&gt;</code> elements with a <code>target</code> attribute value equal to <code>"_blank"</code>
<code>\$(":button")</code>	Selects all <code>&lt;button&gt;</code> elements and <code>&lt;input&gt;</code> elements of <code>type="button"</code>
<code>\$("tr:even")</code>	Selects all even <code>&lt;tr&gt;</code> elements
<code>\$("tr:odd")</code>	Selects all odd <code>&lt;tr&gt;</code> elements

# Will jQuery work with all browsers?

- The jQuery team knows all about cross-browser issues, and they have written this knowledge into the jQuery library.
- jQuery will run exactly the same in all major browsers.



# Common dom events

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

```
<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js">
</script>
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("#div1").load("demo.txt");
  });
});
</script>
</head>
<body>

<div id="div1"><h2>Let jQuery AJAX Change This Text</h2></div>

<button>Get External Content</button>

</body>
</html>
```

# Ajax with jQuery

- AJAX = Asynchronous JavaScript and XML.
- In short; AJAX is about loading data in the background and display it on the webpage, without reloading the whole page.
- 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!

# jQuery load method

- The jQuery load() method is a simple, but powerful AJAX method.
- The load() method loads data from a server and puts the returned data into the selected element.

## Example

```
$("#button").click(function(){
    $("#div1").load("demo_test.txt", function(responseTxt, statusTxt, xhr){
        if(statusTxt == "success")
            alert("External content loaded successfully!");
        if(statusTxt == "error")
            alert("Error: " + xhr.status + ": " + xhr.statusText);
    });
});
```

# jQuery get and post method

- The jQuery `get()` and `post()` methods are used to request data from the server with an HTTP GET or POST request.
- GET - Requests data from a specified resource
- POST - Submits data to be processed to a specified resource

# Ajax with jQuery

```
<script type="text/javascript">  
  $(document).ready(function(event) { ①  
    ② $.ajax({  
      type : "GET",  
      url : remote_url, ③  
      success : function(result) {  
        ④ //process the result  
      },  
      error : function() {  
        //handle error ⑤  
      }  
    })  
  })  
</script>
```