

jQuery

Functionalities & Ajax

Yash Lathiya

|  |  |  |  |
| --- | --- | --- | --- |
| Sr No. | Topic | Date | Page No. |
|  |  |  |  |
| 1 | jQuery Introduction | 04/08/2023 | 2 |
| 2 | Events of jQuery | 04/08/2023 | 6 |
| 3 | jQuery Validation | 07/08/2023 | 9 |
| 4 | jQuery Functions | 07/08/2023 | 10 |
| 5 | Ajax | 07/08/2023 | 13 |

Table of Contents

jQuery Introduction

* jQuery is lightweight javascript library, which minimizes the code to write in javascript.
* Provides pre-defined methods which can concise the code to write in javascript.
* jQuery is translated to javascript, and that particular script is executed during implementation in javscript.
* Execution speed of jQuery is little slower than javascript due to translation process.
* jQuery is easy to understand as well as easy to implement, that’s why it is widely used across the globe.

How to include jQuery

* Download library from jQuery.com
* Include jQuery from CDN

<script

src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js">

</script>

Syntax & jQuery Selector

$(selector).action

* $(this).hide() - hides the current element.
* $("p").hide() - hides all <p> elements.
* $(".test").hide() - hides all elements with class="test".
* $("#test").hide() - hides the element with id="test".
* $(document).ready( //logic );

This method is used to run jQuey after webpage loads.

* jQuery selector properties are similar to selector properties in CSS.
* $(“\*”) 🡪 to select all elements.

HTML / CSS Methods of jQuery

* List of methods are described below which are use to manipulate HTML & CSS content by using jQuery :
* Image showed below represents element dimension understanding in jQuery.



|  |  |
| --- | --- |
| **Method** | **Description** |
| [addClass()](https://www.w3schools.com/jquery/html_addclass.asp) | Adds one or more class names to selected elements |
| [after()](https://www.w3schools.com/jquery/html_after.asp) | Inserts content after selected elements |
| [append()](https://www.w3schools.com/jquery/html_append.asp) | Inserts content at the end of selected elements |
| [appendTo()](https://www.w3schools.com/jquery/html_appendto.asp) | Inserts HTML elements at the end of selected elements |
| [attr()](https://www.w3schools.com/jquery/html_attr.asp) | Sets or returns attributes/values of selected elements |
| [before()](https://www.w3schools.com/jquery/html_before.asp) | Inserts content before selected elements |
| [clone()](https://www.w3schools.com/jquery/html_clone.asp) | Makes a copy of selected elements |
| [css()](https://www.w3schools.com/jquery/css_css.asp) | Sets or returns one or more style properties for selected elements |
| [detach()](https://www.w3schools.com/jquery/html_detach.asp) | Removes selected elements (keeps data and events) |
| [empty()](https://www.w3schools.com/jquery/html_empty.asp) | Removes all child nodes and content from selected elements |
| [hasClass()](https://www.w3schools.com/jquery/html_hasclass.asp) | Checks if any of the selected elements have a specified class name |
| [height()](https://www.w3schools.com/jquery/css_height.asp) | Sets or returns the height of selected elements |
| [html()](https://www.w3schools.com/jquery/html_html.asp) | Sets or returns the content of selected elements |
| [innerHeight()](https://www.w3schools.com/jquery/html_innerheight.asp) | Returns the height of an element (includes padding, but not border) |
| [innerWidth()](https://www.w3schools.com/jquery/html_innerwidth.asp) | Returns the width of an element (includes padding, but not border) |
| [insertAfter()](https://www.w3schools.com/jquery/html_insertafter.asp) | Inserts HTML elements after selected elements |
| [insertBefore()](https://www.w3schools.com/jquery/html_insertbefore.asp) | Inserts HTML elements before selected elements |
| [outerHeight()](https://www.w3schools.com/jquery/html_outerheight.asp) | Returns the height of an element (includes padding and border) |
| [outerWidth()](https://www.w3schools.com/jquery/html_outerwidth.asp) | Returns the width of an element (includes padding and border) |
| [position()](https://www.w3schools.com/jquery/css_position.asp) | Returns the position (relative to the parent element) of an element |
| [prepend()](https://www.w3schools.com/jquery/html_prepend.asp) | Inserts content at the beginning of selected elements |
| [prependTo()](https://www.w3schools.com/jquery/html_prependto.asp) | Inserts HTML elements at the beginning of selected elements |
| [prop()](https://www.w3schools.com/jquery/html_prop.asp) | Sets or returns properties/values of selected elements |
| [remove()](https://www.w3schools.com/jquery/html_remove.asp) | Removes the selected elements (including data and events) |
| [removeAttr()](https://www.w3schools.com/jquery/html_removeattr.asp) | Removes one or more attributes from selected elements |
| [removeClass()](https://www.w3schools.com/jquery/html_removeclass.asp) | Removes one or more classes from selected elements |
| [removeProp()](https://www.w3schools.com/jquery/html_removeprop.asp) | Removes a property set by the prop() method |
| [replaceAll()](https://www.w3schools.com/jquery/html_replaceall.asp) | Replaces selected elements with new HTML elements |
| [replaceWith()](https://www.w3schools.com/jquery/html_replacewith.asp) | Replaces selected elements with new content |
| [text()](https://www.w3schools.com/jquery/html_text.asp) | Sets or returns the text content of selected elements |
| [toggleClass()](https://www.w3schools.com/jquery/html_toggleclass.asp) | Toggles between adding/removing one or more classes from selected elements |
| [unwrap()](https://www.w3schools.com/jquery/html_unwrap.asp) | Removes the parent element of the selected elements |
| [val()](https://www.w3schools.com/jquery/html_val.asp) | Sets or returns the value attribute of the selected elements (for form elements) |
| [width()](https://www.w3schools.com/jquery/css_width.asp) | Sets or returns the width of selected elements |
| wrap() | Wraps HTML element(s) around each selected element |
| [wrapAll()](https://www.w3schools.com/jquery/html_wrapall.asp) | Wraps HTML element(s) around all selected elements |
| [wrapInner()](https://www.w3schools.com/jquery/html_wrapinner.asp) | Wraps HTML element(s) around the content of each selected element |

Events of jQuery

* There are numerous events in jQuery, which is referenced below :

|  |  |
| --- | --- |
| **Method / Property** | **Description** |
| [blur()](https://www.w3schools.com/jquery/event_blur.asp) | Attaches/Triggers the blur event |
| [change()](https://www.w3schools.com/jquery/event_change.asp) | Attaches/Triggers the change event |
| [click()](https://www.w3schools.com/jquery/event_click.asp) | Attaches/Triggers the click event |
| [dblclick()](https://www.w3schools.com/jquery/event_dblclick.asp) | Attaches/Triggers the double click event |
| [event.currentTarget](https://www.w3schools.com/jquery/event_currenttarget.asp) | The current DOM element within the event bubbling phase |
| [event.data](https://www.w3schools.com/jquery/event_data.asp) | Contains the optional data passed to an event method when the current executing handler is bound |
| event.pageX | Returns the mouse position relative to the left edge of the document |
| [event.pageY](https://www.w3schools.com/jquery/event_pagey.asp) | Returns the mouse position relative to the top edge of the document |
| [event.result](https://www.w3schools.com/jquery/event_result.asp) | Contains the last/previous value returned by an event handler triggered by the specified event |
| [event.target](https://www.w3schools.com/jquery/event_target.asp) | Returns which DOM element triggered the event |
| [event.timeStamp](https://www.w3schools.com/jquery/event_timestamp.asp) | Returns the number of milliseconds since January 1, 1970, when the event is triggered |
| [focus()](https://www.w3schools.com/jquery/event_focus.asp) | Attaches/Triggers the focus event |
| [focusin()](https://www.w3schools.com/jquery/event_focusin.asp) | Attaches an event handler to the focusin event |
| [focusout()](https://www.w3schools.com/jquery/event_focusout.asp) | Attaches an event handler to the focusout event |
| [hover()](https://www.w3schools.com/jquery/event_hover.asp) | Attaches two event handlers to the hover event |
| [keydown()](https://www.w3schools.com/jquery/event_keydown.asp) | Attaches/Triggers the keydown event |
| [keypress()](https://www.w3schools.com/jquery/event_keypress.asp) | Attaches/Triggers the keypress event |
| [keyup()](https://www.w3schools.com/jquery/event_keyup.asp) | Attaches/Triggers the keyup event |
| [mousedown()](https://www.w3schools.com/jquery/event_mousedown.asp) | Attaches/Triggers the mousedown event |
| [mouseenter()](https://www.w3schools.com/jquery/event_mouseenter.asp) | Attaches/Triggers the mouseenter event |
| [mouseleave()](https://www.w3schools.com/jquery/event_mouseleave.asp) | Attaches/Triggers the mouseleave event |
| [mousemove()](https://www.w3schools.com/jquery/event_mousemove.asp) | Attaches/Triggers the mousemove event |
| [mouseout()](https://www.w3schools.com/jquery/event_mouseout.asp) | Attaches/Triggers the mouseout event |
| [mouseover()](https://www.w3schools.com/jquery/event_mouseover.asp) | Attaches/Triggers the mouseover event |
| [mouseup()](https://www.w3schools.com/jquery/event_mouseup.asp) | Attaches/Triggers the mouseup event |
| [one()](https://www.w3schools.com/jquery/event_one.asp) | Adds one or more event handlers to selected elements. This handler can only be triggered once per element |
| [ready()](https://www.w3schools.com/jquery/event_ready.asp) | Specifies a function to execute when the DOM is fully loaded |
| [resize()](https://www.w3schools.com/jquery/event_resize.asp) | Attaches/Triggers the resize event |
| [scroll()](https://www.w3schools.com/jquery/event_scroll.asp) | Attaches/Triggers the scroll event |
| [select()](https://www.w3schools.com/jquery/event_select.asp) | Attaches/Triggers the select event |
| [submit()](https://www.w3schools.com/jquery/event_submit.asp) | Attaches/Triggers the submit event |

How to fire event programmatically ?

* In jQuery, events can be fired programmatically with custom logic by using trigger event.
* It’ll give same output as natural event fired by user.
* Trigger method is used for that purpose.
* Detailed implementation is presented in demo of jQuery.

jQuery Validation

* jQuery validation is a validation process of client side.
* It validates data before submitting form.
* It is a good idea to check every field of user input before actually submitting it & acknowledge user to that particular field is valid or not.

jQuery Functions

map()

* map() function maps all data which is given in argument.
* We can give argument in tag, className, idName, :type etc.
* If we want to show all given input in form in p element.. code is here..It prints coma separated values.

$( "p" )

.append( $( "input" ).map(function() {

return $( this ).val();

})

.get()

.join( ", " ) );

grep()

* grep() function finds the element of array which satisfy a filter function, It’ll not affect original array.

var newArray = jQuery.grep(array, function(element, index){

return (element > 5 && index < 10);

});

* This code generates newArray which has applied filter function in array by using grep() method.

extend()

* extend() method is used to merge content of two or more objects together.

**var** object1 = {

apple: 0,

banana: { weight: 52, price: 100 },

cherry: 97

};

**var** object2 = {

banana: { price: 200 },

durian: 100

};

$.extend( object1, object2 );

* extend() method can be written in many more forms to get different results:

$.extend( true, object1, object2 );

**var** object3 = $.extend( {}, object1, object2 );

each()

* each() method is used to iterate over both arrays and objects.

jQuery.each( array, function( index, value) {

//custom logic on index & value

//will stop running after 3.

return (value !== 3);

});

* It also same works in objects, it will consider propertyName as index and propertyValue as value.

merge()

* merge() method is merges the content of two arrays in first array.
* Similar working as appending.

//It will merge values of array2 and array1 in array1.

//array1 is affected.

$merge (array1, array2);

* There are several more ways to use merge() so first array is not getting affected.

var newArray = $.merge([] , oldArray);

$.merge( $.merge( [], firstArray), secondArray );

callbacks()

* It is a multipurpose callbacks list object that provides powerful way to manage callback lists.
* It has following methods to manage callbacaks.

callbacks.add( functionName );

callbacks.remove( functionName );

callbacks.fire( functionName );

callbacks.disable( functionName );

Deferred & Promise Object

* Deferred object is used to create & manage asynchronous operation.
* While promise object is used to access the results of those operations.
* Deferred objects are representation of an operation that may not have been completed yet.

var objectName = $.Deffered();

objectName.resolve();

* Promise object is a representation of the result of a defereed operation, It allows to access the result once deferred object is resolved.

var promise = objectName.promise();

promise.done( () => {} );

promise.fail( () => {} );

* When deferred object is resolved, function within promise.done() will be executed.
* When deferred object is rejected, function within promise.fail() will be executed.

AJAX

Introduction & Use of AJAX

* AJAX is abbreviated form of Asynchronous Javascript and XML.
* AJAX provides facility to exchange data with servers and load them in particular component of webpage without reloading the webpage.
* AJAX loads data in background.
* Typical coding of AJAX is little bit tough and single logic of code doesn’t reach to every browser, It may require different code for support of different browser.
* AJAX implementation with jQuery is quite easy & there is no need to code for different browser.

Send Data with AJAX

load()

* Load method is used to load data from server side.

$(selector).load( URL, data, callback);

$(“#divEmployee”).load( “employee.txt” ,

function(responseTxt, statusTxt, xhr){

if(statusTxt == “success”){

//Code at success

}

if(statusTxt == “error”){

//Code at failure

}

});

get()

* get() method is used to request data from server with HTTP Get request.

$.get( “URL” , function( data , status )){

alert(data + status);

}

post()

$.post(“URL” , { name : “yash” , college : “GEC Rajkot”},

function(data , status) {

alert(data + status);

})

JSON

* JSON stands for javascript object notation.
* Json file is a string which contains data as object format.

{“employee” : [{ “name” : “yash”, “college” : “GEC” },

“name” : “satyam”, “college” : “GEC”]}

serialize()

* This method is used to convert form data into URL encoded text string.

$(“form”).serialize();

deserialize()

* Converts query strings into objects & deserialize form easily.