**JQuery**

**What is jQuery and how jQuery works?**

The 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. jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

The jQuery library contains the following features:

* HTML/DOM manipulation
* CSS manipulation
* HTML event methods
* Effects and animations
* AJAX
* Utilities

The jQuery syntax is tailor made for selecting HTML elements and performing some action on the element(s).

**Basic syntax is: $(selector).action()**

* A $ sign to define/access jQuery
* A (selector) to "query (or find)" HTML elements
* A jQuery action() to be performed on the element(s)

# Document Ready Event

You might have noticed that all jQuery methods in our examples, are inside a document ready event:

$(document).ready(function()

{

//jQuerymethod

});

**Jquery Selectors**

jQuery selectors allow you to select and manipulate HTML element(s). jQuery selectors are used to"find" (or select) HTML elements based on their id, classes, types, attributes, values of attributes and much more. It's based on the existing CSS Selectors, and in addition, it has some own custom selectors. All selectors in jQuery start with the dollar sign and parentheses: $(). 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.

**i) The element Selector**

The jQuery element selector selects elements based on the element name.You can select all

elements on a page like this.

$("p")

**Example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

**<script>**

**$(document).ready(function(){**

**$("button").click(function(){**

**$("p").hide();**

**})**

**});**

**</script>**

</head>

<body>

<h2>This is a heading</h2>

<p>This is a paragraph.</p>

<p>This is another paragraph.</p>

<button>Click me to hide paragraphs</button>

</body>

</html>

**ii) The #id Selector**

The jQuery #id selector uses the id attribute of an HTML tag to find the specific element. An id should be unique within a page, so you should use the #id selector when you want to find a single, unique element.

To find an element with a specific id, write a hash character, followed by the id of the element:$("#test")

**Example**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

**<script>**

**$(document).ready(function(){**

**$("button").click(function(){**

**$("#test").hide();**

**});**

**});**

**</script>**

</head>

<body>

<h2>This is a heading</h2>

<p>This is a paragraph.</p>

<p id="test">This is another paragraph.</p>

<button>Click me</button>

</body>

</html>

iii) The .class Selector

The jQuery class selector finds elements with a specific class. To find elements with a specific class, write a period character, followed by the name of the class: $(".test")

**Example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

**$(document).ready(function(){**

**$("button").click(function(){**

**$(".test").hide();**

**});**

**});**

**</script>**

**</head>**

<body>

<h2 class="test">This is a heading</h2>

<p class="test">This is a paragraph.</p>

<p>This is another paragraph.</p>

<button>Click me</button>

</body>

</html>

## **jQuery Event Methods**

Event methods trigger or attach a function to an event handler for the selected elements.

The following table lists all the jQuery methods used to handle events.

|  |  |
| --- | --- |
| Method / Property | Description |
| bind() | Use the [on()](https://www.w3schools.com/jquery/event_on.asp) method instead. Attaches event handlers to elements |
| blur() | Attaches/Triggers the blur event |
| change() | Attaches/Triggers the change event |
| click() | Attaches/Triggers the click event |
| dblclick() | Attaches/Triggers the double click event |
| delegate() | Deprecate Use the [on()](https://www.w3schools.com/jquery/event_on.asp) method instead. Attaches a handler to current, or future, specified child elements of the matching elements |
| die() | Removed in version 1.9. Removes all event handlers added with the live() method |
| error() | Removed in version 3.0. Attaches/Triggers the error event |
| event.currentTarget | The current DOM element within the event bubbling phase |
| event.data | Contains the optional data passed to an event method when the current executing handler is bound |
| event.delegateTarget | Returns the element where the currently-called jQuery event handler was attached |
| event.isDefaultPrevented() | Returns whether event.preventDefault() was called for the event object |
| event.isImmediatePropagationStopped() | Returns whether event.stopImmediatePropagation() was called for the event object |
| event.isPropagationStopped() | Returns whether event.stopPropagation() was called for the event object |
| event.namespace | Returns the namespace specified when the event was triggered |
| event.pageX | Returns the mouse position relative to the left edge of the document |
| event.pageY | Returns the mouse position relative to the top edge of the document |
| event.preventDefault() | Prevents the default action of the event |
| event.relatedTarget | Returns which element being entered or exited on mouse movement |
| event.result | Contains the last/previous value returned by an event handler triggered by the specified event |
| event.stopImmediatePropagation() | Prevents other event handlers from being called |
| event.stopPropagation() | Prevents the event from bubbling up the DOM tree, preventing any parent handlers from being notified of the event |
| event.target | Returns which DOM element triggered the event |
| event.timeStamp | Returns the number of milliseconds since January 1, 1970, when the event is triggered |
| event.type | Returns which event type was triggered |
| event.which | Returns which keyboard key or mouse button was pressed for the event |
| focus() | Attaches/Triggers the focus event |
| focusin() | Attaches an event handler to the focusin event |
| focusout() | Attaches an event handler to the focusout event |
| hover() | Attaches two event handlers to the hover event |
| keydown() | Attaches/Triggers the keydown event |
| keypress() | Attaches/Triggers the keypress event |
| keyup() | Attaches/Triggers the keyup event |
| live() | Removed in version 1.9. Adds one or more event handlers to current, or future, selected elements |
| load() | Removed in version 3.0. Attaches an event handler to the load event |
| mousedown() | Attaches/Triggers the mousedown event |
| mouseenter() | Attaches/Triggers the mouseenter event |
| mouseleave() | Attaches/Triggers the mouseleave event |
| mousemove() | Attaches/Triggers the mousemove event |
| mouseout() | Attaches/Triggers the mouseout event |
| mouseover() | Attaches/Triggers the mouseover event |
| mouseup() | Attaches/Triggers the mouseup event |
| off() | Removes event handlers attached with the on() method |
| on() | Attaches event handlers to elements |
| one() | Adds one or more event handlers to selected elements. This handler can only be triggered once per element |
| $.proxy() | Takes an existing function and returns a new one with a particular context |
| ready() | Specifies a function to execute when the DOM is fully loaded |
| resize() | Attaches/Triggers the resize event |
| scroll() | Attaches/Triggers the scroll event |
| select() | Attaches/Triggers the select event |
| submit() | Attaches/Triggers the submit event |
| toggle() | Removed in version 1.9. Attaches two or more functions to toggle between for the click event |
| trigger() | Triggers all events bound to the selected elements |
| triggerHandler() | Triggers all functions bound to a specified event for the selected elements |
| unbind() | Deprecated in version 3.0. Use the [off()](https://www.w3schools.com/jquery/event_off.asp) method instead. Removes an added event handler from selected elements |
| undelegate() | Deprecated in version 3.0. Use the [off()](https://www.w3schools.com/jquery/event_off.asp) method instead. Removes an event handler to selected elements, now or in the future |
| unload() | Removed in version 3.0. Use the [on()](https://www.w3schools.com/jquery/event_on.asp) or [trigger()](https://www.w3schools.com/jquery/event_trigger.asp) method instead. Attaches an event handler to the unload event |

jQuery Fading Methods

With jQuery you can fade an element in and out of visibility.

jQuery has the following fade methods:

* fadeIn()
* fadeOut()
* fadeToggle()
* fadeTo()

## jQuery fadeIn() Method

The jQuery fadeIn() method is used to fade in a hidden element.

**Syntax:**

$(selector).fadeIn(speed,callback);

The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.

The optional callback parameter is a function to be executed after the fading completes.

The following example demonstrates the fadeIn() method with different parameters:

### **Example**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("#div1").fadeIn();

$("#div2").fadeIn("slow");

$("#div3").fadeIn(3000);

});

});

</script>

</head>

<body>

<p>Demonstrate fadeIn() with different parameters.</p>

<button>Click to fade in boxes</button><br><br>

<div id="div1" style="width:80px;height:80px;display:none;background-color:red;"></div><br>

<div id="div2" style="width:80px;height:80px;display:none;background-color:green;"></div><br>

<div id="div3" style="width:80px;height:80px;display:none;background-color:blue;"></div>

</body>

</html>

## **jQuery fadeOut() Method**

The jQuery fadeOut() method is used to fade out a visible element.

**Syntax:**

$(selector).fadeOut(speed,callback);

The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.

The optional callback parameter is a function to be executed after the fading completes.

The following example demonstrates the fadeOut() method with different parameters:

**Example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("#div1").fadeOut();

$("#div2").fadeOut("slow");

$("#div3").fadeOut(3000);

});

});

</script>

</head>

<body>

<p>Demonstrate fadeOut() with different parameters.</p>

<button>Click to fade out boxes</button><br><br>

<div id="div1" style="width:80px;height:80px;background-color:red;"></div><br>

<div id="div2" style="width:80px;height:80px;background-color:green;"></div><br>

<div id="div3" style="width:80px;height:80px;background-color:blue;"></div>

</body>

</html>

**jQuery fadeToggle() Method**

The jQuery fadeToggle() method toggles between the fadeIn() and fadeOut() methods.

If the elements are faded out, fadeToggle() will fade them in.

If the elements are faded in, fadeToggle() will fade them out.

**Syntax:**

$(selector).fadeToggle(speed,callback);

The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.

The optional callback parameter is a function to be executed after the fading completes.

**Example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("#div1").fadeToggle();

$("#div2").fadeToggle("slow");

$("#div3").fadeToggle(3000);

});

});

</script>

</head>

<body>

<p>Demonstrate fadeToggle() with different speed parameters.</p>

<button>Click to fade in/out boxes</button><br><br>

<div id="div1" style="width:80px;height:80px;background-color:red;"></div>

<br>

<div id="div2" style="width:80px;height:80px;background-color:green;"></div>

<br>

<div id="div3" style="width:80px;height:80px;background-color:blue;"></div>

</body>

</html>

## **jQuery fadeTo() Method**

The jQuery fadeTo() method allows fading to a given opacity (value between 0 and 1).

**Syntax:**

$(selector).fadeTo(speed,opacity,callback);

The required speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.

The required opacity parameter in the fadeTo() method specifies fading to a given opacity (value between 0 and 1).

The optional callback parameter is a function to be executed after the function completes.

**Example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("#div1").fadeTo("slow", 0.15);

$("#div2").fadeTo("slow", 0.4);

$("#div3").fadeTo("slow", 0.7);

});

});

</script>

</head>

<body>

<p>Demonstrate fadeTo() with different parameters.</p>

<button>Click to fade boxes</button><br><br>

<div id="div1" style="width:80px;height:80px;background-color:red;"></div><br>

<div id="div2" style="width:80px;height:80px;background-color:green;"></div><br>

<div id="div3" style="width:80px;height:80px;background-color:blue;"></div>

</body>

</html>

## **jQuery Sliding Methods**

With jQuery you can create a sliding effect on elements.

jQuery has the following slide methods:

* slideDown()
* slideUp()
* slideToggle()

**jQuery slideDown() Method**

The jQuery slideDown() method is used to slide down an element.

**Syntax:**

$(selector).slideDown(speed,callback);

The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.

The optional callback parameter is a function to be executed after the sliding completes.

The following example demonstrates the slideDown() method:

### **Example**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("#flip").click(function(){

$("#panel").slideDown("slow");

});

});

</script>

<style>

#panel, #flip {

padding: 5px;

text-align: center;

background-color: #e5eecc;

border: solid 1px #c3c3c3;

}

#panel {

padding: 50px;

display: none;

}

</style>

</head>

<body>

<div id="flip">Click to slide down panel</div>

<div id="panel">Hello world!</div>

</body>

</html>

## **jQuery slideUp() Method**

The jQuery slideUp() method is used to slide up an element.

**Syntax:**

$(selector).slideUp(speed,callback);

The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.

The optional callback parameter is a function to be executed after the sliding completes.

**Example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("#flip").click(function(){

$("#panel").slideUp("slow");

});

});

</script>

<style>

#panel, #flip {

padding: 5px;

text-align: center;

background-color: #e5eecc;

border: solid 1px #c3c3c3;

}

#panel {

padding: 50px;

}

</style>

</head>

<body>

<div id="flip">Click to slide up panel</div>

<div id="panel">Hello world!</div>

</body>

</html>

## **jQuery slideToggle() Method**

The jQuery slideToggle() method toggles between the slideDown() and slideUp() methods.

If the elements have been slid down, slideToggle() will slide them up.

If the elements have been slid up, slideToggle() will slide them down.

$(selector).slideToggle(speed,callback);

The optional speed parameter can take the following values: "slow", "fast", milliseconds.

The optional callback parameter is a function to be executed after the sliding completes.

**Example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("#flip").click(function(){

$("#panel").slideToggle("slow");

});

});

</script>

<style>

#panel, #flip {

padding: 5px;

text-align: center;

background-color: #e5eecc;

border: solid 1px #c3c3c3;

}

#panel {

padding: 50px;

display: none;

}

</style>

</head>

<body>

<div id="flip">Click to slide the panel down or up</div>

<div id="panel">Hello world!</div>

</body>

</html>

**1.DB Click**

**example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("p").dblclick(function(){

$(this).hide();

});

});

</script>

</head>

<body>

<p>If you double-click on me, I will disappear.</p>

<p>Click me away!</p>

<p>Click me too!</p>

</body>

</html>

**2.MouseEnter**

**example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("#p1").mouseenter(function(){

alert("You entered p1!");

});

});

</script>

</head>

<body>

<p id="p1">Enter this paragraph.</p>

</body>

</html>

**3.MouseLeave()**

**Example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("#p1").mouseleave(function(){

alert("Bye! You now leave p1!");

});

});

</script>

</head>

<body>

<p id="p1">This is a paragraph.</p>

</body>

</html>

**4.Hover()**

**Example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("#p1").hover(function(){

alert("You entered p1!");

},

function(){

alert("Bye! You now leave p1!");

});

});

</script>

</head>

<body>

<p id="p1">This is a paragraph.</p>

</body>

</html>

**5.Focus()**

**Example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("input").focus(function(){

$(this).css("background-color", "yellow");

});

$("input").blur(function(){

$(this).css("background-color", "green");

});

});

</script>

</head>

<body>

Name: <input type="text" name="fullname"><br>

Email: <input type="text" name="email">

</body>

</html>

**6.Blur()**

**Example:**

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("input").focus(function(){

$(this).css("background-color", "yellow");

});

$("input").blur(function(){

$(this).css("background-color", "green");

});

});

</script>

</head>

<body>

Name: <input type="text" name="fullname"><br>

Email: <input type="text" name="email">

</body>

</html>