

jQuery

jQuery is a fast, small, cross-platform and feature-rich JavaScript library. It is designed to simplify the client-side scripting of HTML. It makes things like HTML document traversal and manipulation, animation, event handling, and AJAX very simple with an easy-to-use API that works on a lot of different type of browsers.

The main purpose of jQuery is to provide an easy way to use JavaScript on your website to make it more interactive and attractive. It is also used to add animation.

What is jQuery

jQuery is a small, light-weight and fast JavaScript library. It is cross-platform and supports different types of browsers. It is also referred as "write less do more" because it takes a lot of common tasks that requires many lines of JavaScript code to accomplish, and binds them into methods that can be called with a single line of code whenever needed. It is also very useful to simplify a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

- jQuery is a small, fast and lightweight JavaScript library.
- jQuery is platform-independent.
- jQuery means "write less do more".
- jQuery simplifies AJAX call and DOM manipulation.

jQuery Features

Following are the important features of jQuery.

- HTML manipulation
- DOM manipulation
- DOM element selection
- CSS manipulation
- Effects and Animations
- Utilities
- AJAX
- HTML event methods
- JSON Parsing
- Extensibility through plug-ins

Why jQuery is required

Sometimes, a question can arise that what is the need of jQuery or what difference it makes on bringing jQuery instead of AJAX/ JavaScript? If jQuery is the replacement of AJAX and JavaScript? For all these questions, you can state the following answers.

- It is very fast and extensible.
- It facilitates the users to write UI related function codes in minimum possible lines.
- It improves the performance of an application.
- Browser's compatible web applications can be developed.
- It uses mostly new features of new browsers.

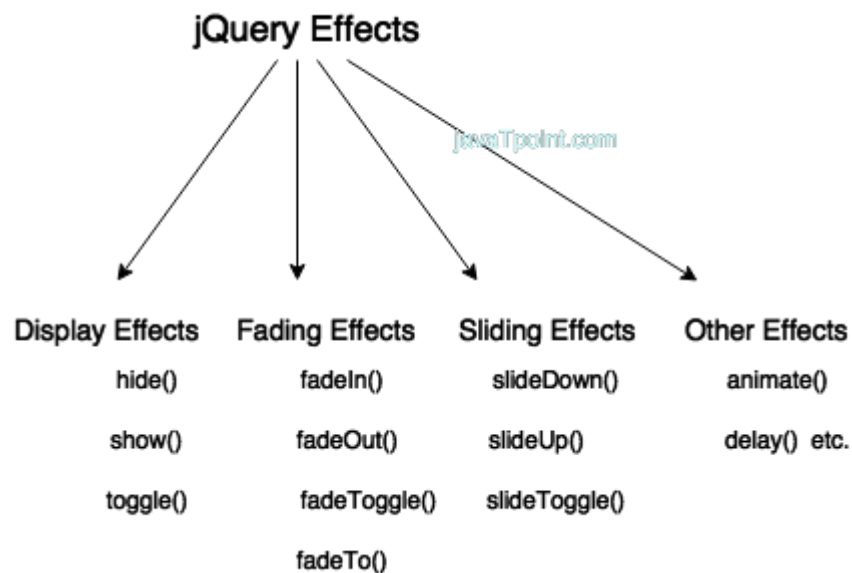
So, you can say that out of the lot of JavaScript frameworks, jQuery is the most popular and the most extendable. Many of the biggest companies on the web use jQuery.

Some of these companies are:

- Microsoft
- Google
- IBM
- Netflix

jQuery Effects

jQuery enables us to add effects on a web page. jQuery effects can be categorized into fading, sliding, hiding/showing and animation effects.



jQuery hide()

The jQuery hide() method is used to hide the selected elements.

Syntax:

1. \$(selector).hide();
2. \$(selector).hide(speed, callback);
3. \$(selector).hide(speed, easing, callback);

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
5. <script>
6. \$(document).ready(function(){
7. \$("#hide").click(function(){
8. \$("p").hide();
9. });
10. });
11. </script>
12. </head>
13. <body>
14. <p>
15. This is a little poem:

16. Twinkle, twinkle, little star

17. How I wonder what you are

18. Up above the world so high

19. Like a diamond in the sky

20. Twinkle, twinkle little star

21. How I wonder what you are
22. </p>
23. <button id="hide">Hide</button>
24. </body>
25. </html>

jQuery show()

The jQuery show() method is used to show the selected elements.

Syntax:

1. `$(selector).show();`
2. `$(selector).show(speed, callback);`
3. `$(selector).show(speed, easing, callback);`

1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`
4. `<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>`
5. `<script>`
6. `$(document).ready(function(){`
7. `$("#hide").click(function(){`
8. `$("p").hide();`
9. `});`
10. `$("#show").click(function(){`
11. `$("p").show();`
12. `});`
13. `});`
14. `</script>`
15. `</head>`
16. `<body>`
17. `<p>`
18. `This is a little poem:
`
19. `Twinkle, twinkle, little star
`
20. `How I wonder what you are
`
21. `Up above the world so high
`
22. `Like a diamond in the sky
`
23. `Twinkle, twinkle little star
`
24. `How I wonder what you are`
25. `</p>`
26. `<button id="hide">Hide</button>`
27. `<button id="show">Show</button>`
28. `</body>`
29. `</html>`

jQuery toggle()

The jQuery toggle() is a special type of method which is used to toggle between the hide() and show() method. It shows the hidden elements and hides the shown element.

Syntax:

1. `$(selector).toggle();`
2. `$(selector).toggle(speed, callback);`
3. `$(selector).toggle(speed, easing, callback);`
4. `$(selector).toggle(display);`

1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`
4. `<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>`
5. `<script>`
6. `$(document).ready(function(){`
7. `$("button").click(function(){`
8. `$("div.d1").toggle();`
9. `});`
10. `});`
11. `</script>`
12. `</head>`
13. `<body>`
14. `<button>Toggle</button>`
15. `<div class="d1" style="border:1px solid black;padding:10px;width:250px">`
16. `<p>This is a little poem:
`
17. `Twinkle, twinkle, little star
`
18. `How I wonder what you are
`
19. `Up above the world so high
`
20. `Like a diamond in the sky
`
21. `Twinkle, twinkle little star
`
22. `How I wonder what you are</p>`
23. `</div>`
24. `</body>`
25. `</html>`

jQuery html()

jQuery `html()` method is used to change the entire content of the selected elements. It replaces the selected element content with new contents.

Note: It is a very useful function but works in a limited area because of its API documentation. The API documentation of the jQuery html function consists of three method signatures.

The first method signature has no argument, so it just returns the HTML within that element. The remaining two signatures take a single argument: i.e. a string or a function that returns a string.

Syntax:

1. `$(selector).html()`

It is used to return content.

1. `$(selector).html(content)`

It is used to set content.

1. `$(selector).html(function (index, currentcontent))`

It is used to set content by calling function.

1. `<html>`
2. `<head>`
3. `<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>`
4. `<script>`
5. `$(document).ready(function(){`
6. `$("button").click(function(){`
7. `alert($("#p").html());`
8. `});`
9. `});`
10. `</script>`
11. `</head>`
12. `<body>`
- 13.
14. `<button>Return the content of p element</button>`
- 15.
16. `<p>This is first paragraph.</p>`
17. `<p>This is another paragraph.</p>`
18. `</body>`
19. `</html>`

jQuery val()

There are two usage of jQuery val() method.

- It is used to get current value of the first element in the set of matched elements.

- It is used to set the value of every matched element.

Syntax:

1. `$(selector).val()`

It is used to get value.

1. `$(selector).val(value)`

It is used to set value.

1. `$(selector).val(function(index,currentvalue))`

It is used to set value using function.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>Using jquery</title>
```

```
    <script src="jquery/jquery-3.3.1.min.js"></script>
```

```
    <script type="text/javascript" language="javascript">
```

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

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

```
                var str=$("#txt").val();
```

```
                $("p").html("Hello <h2> "+str+"</h2>");
```

```
            });
```

```
        });
```

```
    </script>
```

```
    <style type="text/css">
```

```
        .p{height:200px;width:400px;background-color:cyan;margin-top:20px;position:absolute;}
```

```
    </style>
```

```
</head>
```

```
<body>
```

```
<button>Click here to change the content of all p elements</button>
```

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

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

Type your Message

<input type="text" id="txt"/>

</body>

</html>

jQuery text()

The jQuery text() method is used to set or return the text content of the selected elements.

To return content: When this method is used to return content, it returns the combined text content of all matched elements without the HTML markup.

To set content: When this method is used to set content, it overwrites the content of all matched elements.

Difference between jQuery text() method and jQuery html() method

Sometimes, this confusion is occurred because both of the methods are used to set or return the html content. But, the jQuery text() method is different from html() method.

Following is the main differences:

- The jQuery text() method is used to set or return html content without HTML markup while, html() method is used to set or return the innerHtml (text + HTML markup).
- The jQuery text() method can be used in both XML and HTML document while jQuery html() method can't.

Syntax:

To return text content:

1. \$(selector).text()

To set text content:

1. \$(selector).text(content)

Example of jQuery text() method

Let's take an example to demonstrate the effect of jQuery text() method.

A simple example to **return** content:

1. <!DOCTYPE html>
2. <html lang="en">

3. `<head>`
4. `<meta charset="utf-8">`
5. `<title>text demo</title>`
6. `<script src="https://code.jquery.com/jquery-1.10.2.js"></script>`
7. `</head>`
8. `<body>`
9. `<p>Hello! javatpoint.com</p>`
10. `<p></p>`
11. `<script>`
12. `var str = $("p:first").text();`
13. `$("p:last").html(str);`
14. `</script>`
15. `</body>`
16. `</html>`

example of jQuery text() method

An example for **set** content:

1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`
4. `<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>`
5. `<script>`
6. `$(document).ready(function(){`
7. `$("button").click(function(){`
8. `$("p").text("Welcome to javatpoint.com!");`
9. `});`
10. `});`
11. `</script>`
12. `</head>`
13. `<body>`
14. `<button>Click here to set text content for all p elements</button>`
15. `<p>Hello Guys!</p>`
16. `<p>Looking for online training....</p>`
17. `</body>`
18. `</html>`

jQuery css()

The jQuery CSS() method is used to get (return) or set style properties or values for selected elements. It facilitates you to get one or more style properties.

jQuery CSS() method provides two ways:

1) Return a CSS property

It is used to get the value of a specified CSS property.

Syntax:

1. `css("propertyname");`

Let's take an example to demonstrate this property.

1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`
4. `<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>`
5. `<script>`
6. `$(document).ready(function(){`
7. `$("button").click(function(){`
8. `alert("Background color = " + $("p").css("background-color"));`
9. `});`
10. `});`
11. `</script>`
12. `</head>`
13. `<body>`
14. `<h2>This is a heading</h2>`
15. `<p style="background-color: #ff0000">The background-color of this paragraph is red.</p>`
16. `<p style="background-color: #00ff00">The background-color of this paragraph is green.</p>`
17. `<p style="background-color: #0000ff">The background-color of this paragraph is blue.</p>`
18. `<button>Click here to get the background-color of first matched element</button>`
19. `</body>`
20. `</html>`

2) Set a CSS property

This property is used to set a specific value for all matched element.

Syntax:

1. `css("propertyname","value");`
1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`
4. `<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>`
5. `<script>`
6. `$(document).ready(function(){`
7. `$("button").click(function(){`
8. `$("p").css("background-color", "violet");`
9. `});`
10. `});`
11. `</script>`
12. `</head>`
13. `<body>`
14. `<p style="background-color: #ff0000">The background-color of this paragraph is red.</p>`
15. `<p style="background-color: #00ff00">The background-color of this paragraph is green.</p>`
16. `<p style="background-color: #0000ff">The background-color of this paragraph is blue.</p>`
17. `<p>This paragraph has no background-color. </p>`
18. `<button>Click here to set a specific background-color of all matched element</button>`
19. `</body>`
20. `</html>`

3) Set multiple CSS properties

It is just an extension of Set CSS property. It facilitates you to add multiple property values together.

Syntax:

1. `css({"propertyname":"value","propertyname":"value",...});`
Let's take an example to demonstrate this property. In this example we add two properties background-color and font-size for all element.
1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`

```

4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("button").click(function(){
8.         $("p").css({"background-color": "yellow", "font-size": "200%", "font-style": "italic"});
9.     });
10. });
11. </script>
12. </head>
13. <body>
14. <h2>This is a heading</h2>
15. <p style="background-color: #ff0000">The background-
    color of this paragraph is red.</p>
16. <p style="background-color: #00ff00">The background-
    color of this paragraph is green.</p>
17. <p style="background-color: #0000ff">The background-
    color of this paragraph is blue.</p>
18. <p>This paragraph has no background-color.</p>
19. <button>Click here to set multiple styles for all selected elements.</button>
20. </body>
21. </html>

```

jQuery before()

The jQuery before() method is used to insert the specified content before the selected elements. It adds the content specified by the parameter, before each element in the set of matched elements.

Example of jQuery before() method

Let's take an example to demonstrate the jQuery before() method.

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("button").click(function(){

```

```

8.      $("p").before("<p><b>Hello javatpoint.com</b></p>");
9.    });
10. });
11. </script>
12. </head>
13. <body>
14. <button>Insert content before each p element</button>
15. <p>This is a tutorial website.</p>
16. <p>This is a training institute.</p>
17. </body>
18. </html>

```

jQuery prepend()

The jQuery prepend() method is used to insert the specified content at the beginning (as a first child) of the selected elements. It is just the opposite of the jQuery append() method.

If you want to insert the content at the end of the selected elements, you should use the append method.

Example of jQuery prepend() method

Let's take an example to demonstrate the jQuery prepend() method.

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.   $("#btn1").click(function(){
8.     $("p").prepend("<b>Prepended text</b>. ");
9.   });
10. });
11. </script>
12. </head>
13. <body>
14. <p>This is the first paragraph.</p>
15. <p>This is the second paragraph.</p>
16. <button id="btn1">Prepend text</button>
17. </body>

```

18. `</html>`

jQuery prepend() example 2

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("#btn1").click(function(){
8.         $("p").prepend("<b>Prepended text</b> ");
9.     });
10.    $("#btn2").click(function(){
11.        $("ol").prepend("<li>Prepended item</li>");
12.    });
13.});
14. </script>
15. </head>
16. <body>
17. <p>This is the first paragraph.</p>
18. <p>This is the second paragraph.</p>
19. <ol>
20. <li>Item no.1</li>
21. <li>Item no.2</li>
22. <li>Item no.3</li>
23. </ol>
24. <button id="btn1">Prepend text</button>
25. <button id="btn2">Prepend list item</button>
26. </body>
27. </html>
```

jQuery after()

The jQuery after() method is used to insert specified content after the selected element. It is just like jQuery append() method.

If you want to insert content before the selected element, you should use jQuery before() method.

Example of jQuery after() method

Let's see an example of jQuery after() method:

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("button").click(function(){
8.         $("p").after("<p><b>Hello javatpoint.com</b></p>");
9.     });
10. });
11. </script>
12. </head>
13. <body>
14. <button>Insert content after each p element</button>
15. <p>This is a tutorial website.</p>
16. <p>This is a training institute.</p>
17. </body>
18. </html>
```

jQuery append()

The jQuery append() method is used to insert specified content as the last child (at the end of) the selected elements in the jQuery collection.

The append () and appendTo () methods are used to perform the same task. The only difference between them is in the syntax.

Example of jQuery append() method

Let's take an example to demonstrate the jQuery append() method.

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
```

```

7.    $("#btn1").click(function(){
8.        $("p").append(" <b>Newly added appended text</b>.");
9.    });
10.   $("#btn2").click(function(){
11.       $("ol").append("<li><b>Newly added appended item</b></li>");
12.   });
13.});
14.</script>
15.</head>
16.<body>
17.<p>This is a paragraph.</p>
18.<p>This is another paragraph.</p>
19.<ol>
20.  <li>Item no.1</li>
21.  <li>Item no.2</li>
22.  <li>Item no.3</li>
23.</ol>
24.<button id="btn1">Append text</button>
25.<button id="btn2">Append item</button>
26.</body>
27.</html>

```

jQuery remove()

The jQuery remove() method is used to remove the selected elements out of the DOM. It removes the selected element itself, as well as everything inside it (including all texts and child nodes). This method also removes the data and the events of the selected elements.

jQuery remove() example

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("button").click(function(){
8.         $("p").remove();
9.     });
10.});

```



```
11. </script>
12. </head>
13. <body>
14. <p>Welcome Guys!</p>
15. <p><b>This is javatpoint.com</b></p>
16. <button>Click here to execute remove() method</button>
17. </body>
18. </html>
```

jQuery empty()

The jQuery empty() method is used to remove all child nodes and content from the selected elements. This method doesn't remove the element itself.

Example of jQuery empty() method

Let's take an example to demonstrate the jQuery empty() method.

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("button").click(function(){
8.         $("div").empty();
9.     });
10. });
11. </script>
12. </head>
13. <body>
14. <div style="height:150px;background-color:yellow">
15. Twinkle, twinkle, little star,<br/>
16. How I wonder what you are!<br/>
17. Up above the world so high,<br/>
18. Like a diamond in the sky.<br/>
19. Twinkle, twinkle, little star,<br/>
20. How I wonder what you are!<br/>
21. <p><b>This poem is written inside the div.</b></p>
22. </div>
```

23. `<p>`This paragraph is written outside the div.`</p>`
24. `<button>`Execute empty() method to remove the content of div element.`</button>`
25. `</body>`
26. `</html>`

jQuery scrollTop()

The jQuery scrollTop method is used to set or return the vertical scrollbar position for the selected element. When the scrollbar is on the top, it specifies the position 0.

- **To return the position:** When this method is used to return the position, it provides the current vertical position of the first matched element in the set.
- **To set the position:** When this method is used to set the position, it sets the vertical position of the scrollbar for all matched element.

Example of jQuery scrollTop() method

Let's take an example to demonstrate the jQuery scrollTop() method.

1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`
4. `<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>`
5. `<script>`
6. `$(document).ready(function(){`
7. `$("button").click(function(){`
8. `alert($("#div").scrollTop() + " px");`
9. `});`
10. `});`
11. `</script>`
12. `</head>`
13. `<body>`
14. `<div style="border:1px solid black;width:150px;height:100px;overflow:auto">`
15. Twinkle, twinkle, little star,How I wonder what you are!
16. Up above the world so high, Like a diamond in the sky.
17. Twinkle, twinkle, little star, How I wonder what you are!`</div>
`
18. `<button>`Return the vertical position of the scrollbar`</button>`
19. `<p>`Move the scrollbar down and click the button again.`</p>`
20. `</body>`
21. `</html>`

jQuery attr()

The jQuery attr() method is used to set or return attributes and values of the selected elements.

There are two usage of jQuery attr() method.

1. **To return attribute value:** This method returns the value of the first matched element.
2. **To set attribute value:** This method is used to set one or more attribute/value pairs of the set of matched elements.

Syntax:

To return an attribute's value:

1. \$(selector).attr(attribute)

To set an attribute and value:

1. \$(selector).attr(attribute,value)

To set an attribute and value by using a function:

1. \$(selector).attr(attribute,function(index,currentvalue))

To set multiple attributes and values:

1. \$(selector).attr({attribute:value, attribute:value,...})

Example of jQuery attr() method

Let's take an example to demonstrate jQuery attr() method.

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. \$(document).ready(function(){
7. \$("button").click(function(){
8. \$("img").attr("width", "500");
9. });
10. });
11. </script>
12. </head>
13. <body>

14. `
`
15. `<button>` Set the width attribute of the image `</button>`
16. `</body>`
17. `</html>`

jQuery offset()

The jQuery offset() method is used to get the current offset of the first matched element.

It provides two methods: to set or return the offset co-ordinates for the selected elements, relative to the document.

- **To return the offset:** When this method is used to return the offset, it returns the offset co-ordinates of the FIRST matched element. It specifies the object's two properties: the top and left positions in pixels.
- **To set the offset:** When this method is used to set the offset, it sets the offset co-ordinates of ALL matched elements.

Syntax:

To **RETURN** the offset co-ordinates:

1. `$(selector).offset()`

To **SET** the offset co-ordinates:

1. `$(selector).offset({top:value,left:value})`

Example of jQuery offset() method

Let's take an example to demonstrate the jQuery offset() method.

1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`
4. `<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>`
5. `<script>`
6. `$(document).ready(function(){`
7. `$("button").click(function(){`
8. `var x = $("p").offset();`
9. `alert("Top: " + x.top + " Left: " + x.left);`
10. `});`
11. `});`

```
12. </script>
13. </head>
14. <body>
15. <p>You are reading this tutorial on javatpoint.com</p>
16. <button>Click here to return the offset coordinates of the p element</button>
17. </body>
18. </html>
```

jQuery position()

The jQuery position () method makes you able to retrieve the current position of an element relative to the parent element. It returns the position of the first matched element. This method returns the object with two properties: top and left position in pixels.

Example of jQuery position() method

Let's take an example to demonstrate the jQuery position() method.

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("button").click(function(){
8.         var x = $("p").position();
9.         alert("Top position: " + x.top + " Left position: " + x.left);
10.    });
11. });
12. </script>
13. </head>
14. <body>
15. <p>You are reading this tutorial on javatpoint.com</p>
16. <button>Click here to return the offset coordinates of the p element</button>
17. </body>
18. </html>
```

jQuery addClass()

The `addClass()` method is used to add one or more class name to the selected element. This method is used only to add one or more class names to the class attributes not to remove the existing class attributes.

If you want to add more than one class separate the class names with spaces.

Example of jQuery `addClass()` method

Let's take an example to demonstrate the effect of jQuery `addClass()` method:

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("button").click(function(){
8.         $("p:first").addClass("intro");
9.     });
10. });
11. </script>
12. <style>
13. .intro {
14.     font-size: 200%;
15.     color: red;
16. }
17. </style>
18. </head>
19. <body>
20. <h1>This is a heading</h1>
21. <p>This is a paragraph.</p>
22. <p>This is another paragraph.</p>
23. <button>Add a class name to the first p element</button>
24. </body>
25. </html>
```

jQuery `hasClass()`

The jQuery `hasClass()` method is used to check whether selected elements have specified class name or not. It returns `TRUE` if the specified class is present in any of the selected elements otherwise it returns `FALSE`.

Syntax:

1. `$(selector).hasClass(classname)`

Example of jQuery hasClass() method

Let's take an example to demonstrate jQuery hasClass() method.

1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`
4. `<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>`
5. `<script>`
6. `$(document).ready(function(){`
7. `$("button").click(function(){`
8. `alert($("#p").hasClass("intro"));`
9. `});`
10. `});`
11. `</script>`
12. `<style>`
13. `.intro {`
14. `font-size: 150%;`
15. `color: Blue;`
16. `}`
17. `</style>`
18. `</head>`
19. `<body>`
20. `<h1>Look here, I am a heading.</h1>`
21. `<p class="intro">This is a paragraph.</p>`
22. `<p>This is also a paragraph.</p>`
23. `<button>Click here to check if any p element have an "intro" class?</button>`
24. `</body>`
25. `</html>`

jQuery toggleClass()

The jQuery toggleClass() method is used to add or remove one or more classes from the selected elements. This method toggles between adding and removing one or more class name. It checks each element for the specified class names. If the class name is already set, it removes and if the class name is missing, it adds.

Example of jQuery toggleClass() method

Let's take an example to demonstrate the effect of jQuery toggleClass() method.

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("button").click(function(){
8.         $("p").toggleClass("main");
9.     });
10. });
11. </script>
12. <style>
13. .main {
14.     font-size: 150%;
15.     color: red;
16. }
17. </style>
18. </head>
19. <body>
20. <button>Toggle class "main" for p elements</button>
21. <p>Hello! javatpoint.com</p>
22. <p>This is popular tutorial website.</p>
23. <p><b>Note:</b> Click repeatedly on the button to see the toggle effect.</p>
24. </body>
25. </html>
```

jQuery toggleClass() example 2

```
1. <!DOCTYPE html>
2. <html lang="en">
3. <head>
4. <meta charset="utf-8">
5. <title>toggleClass demo</title>
6. <style>
7. p {
8.     margin: 4px;
```



```
9.    font-size: 16px;
10.   font-weight: bolder;
11.   cursor: pointer;
12. }
13. .blue {
14.   color: black;
15. }
16. .highlight {
17.   background: pink;
18. }
19. </style>
20. <script src="https://code.jquery.com/jquery-1.10.2.js"></script>
21. </head>
22. <body>
23. <p class="blue highlight">www.javatpoint.com</p>
24. <p class="blue">Java Tutorial</p>
25. <p class="blue">SQL Tutorial</p>
26. <p class="blue">Android Tutorial</p>
27. <p class="blue">HTML Tutorial</p>
28. <p class="blue">etc.</p>
29. <script>
30. $( "p" ).click(function() {
31. $( this ).toggleClass( "highlight" );
32. });
33. </script>
34. </body>
35. </html>
```

jQuery wrap()

jQuery wrap() method is used to wrap specified HTML elements around each selected element. The wrap () function can accept any string or object that could be passed through the \$() factory function

Example of jQuery wrap() method

Let's take an example to demonstrate the jQuery wrap() method.

```
1. <!DOCTYPE html>
2. <html>
3. <head>
```

```

4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("button").click(function(){
8.         $("p").wrap("<div></div>");
9.     });
10. });
11. </script>
12. <style>
13. div{background-color: pink;}
14. </style>
15. </head>
16. <body>
17. <p>Hello Guys!</p>
18. <p>This is javatpoint.com</p>
19. <button>Wrap a div element around each p element</button>
20. </body>
21. </html>

```

jQuery wrapInner()

The jQuery wrapInner() method is used to wrap an HTML structure around the content of each element in the set of matched element. This method can accept any string or object that could be passed to the \$() factory function.

Example of jQuery wrapInner() method

Let's take an example to demonstrate the jQuery wrapInner() method.

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("button").click(function(){
8.         $("p").wrapInner("<em></em>");
9.     });
10. });

```

```
11. </script>
12. </head>
13. <body>
14. <p>Hello Guys!</p>
15. <p>This is javatpoint.com</p>
16. <button>Wrap a emphasized element around the content of each p element</button>
17. </body>
18. </html>
```

jQuery wrapInner() example 2

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("button").click(function(){
8.         $("p").wrapInner("<em><b><marquee></marquee></b></em>");
9.     });
10. });
11. </script>
12. </head>
13. <body>
14. <p>Hello Guys!</p>
15. <p>This is javatpoint.com</p>
16. <button>Wrap a emphasized element around the content of each p element</button>
17. </body>
18. </html>
```

jQuery wrapAll()

jQuery wrapAll() method is used to wrap specified HTML elements around all selected elements, in a set of matched elements.

Example of jQuery wrapAll() method

Let's take an example to demonstrate the jQuery wrapAll() method.

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. \$(document).ready(function(){
7. \$("button").click(function(){
8. \$("p").wrapAll("<div></div>");
9. });
10. });
11. </script>
12. <style>
13. div{background-color: pink;}
14. </style>
15. </head>
16. <body>
17. <p>Hello Guys!</p>
18. <p>This is javatpoint.com</p>
19. <button>Wrap a div element around all p elements</button>
20. </body>
21. </html>

jQuery unwrap()

The jQuery unwrap() method is used to remove the parent element of the selected elements.

Example of jQuery unwrap() method

Let's take an example to demonstrate the jQuery unwrap() method.

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. \$(document).ready(function(){
7. \$("button").click(function(){
8. \$("p").unwrap();

```

9.     });
10. });
11. </script>
12. <style>
13. div{background-color: orange;}
14. article{background-color: yellowgreen;}
15. </style>
16. </head>
17. <body>
18. <div>
19. <p>Hello Guys!</p>
20. </div>
21. <article>
22. <p>This is javatpoint.com</p>
23. </article>
24. <button>Click here to remove the parent element of each p element</button>
25. </body>
26. </html>

```

jQuery serialize()

jQuery serialize() method is used to create a text string in standard URL-encoded notation. It is used in form controls like <input>, <textarea>, <select> etc. It serializes the form values so that its serialized values can be used in the URL query string while making an AJAX request.

jQuery serialize() example

Let's take an example which serializes a form values.

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4.   <script src="https://code.jquery.com/jquery-1.10.2.js"></script>
5.   <script>
6.     $(document).ready(function(){
7.       $("button").click(function(){
8.         $("div").text($("#form").serialize());
9.       });
10.    });
11.  </script>
12. </head>

```

```
13. <body>
14. <form action="">
15. First name: <input type="text" name="FirstName" value="Sonoo"><br>
16. Last name: <input type="text" name="LastName" value="Jaiswal"><br>
17. </form>
18. <button>Serialize form values</button>
19. <div></div>
20. </body>
21. </html>
```

jQuery bind()

The jQuery bind() event is used to attach one or more event handlers for selected elements from a set of elements. It specifies a function to run when the event occurs.

It is generally used together with other events of jQuery.

Example of jQuery bind() event

Let's take an example to demonstrate jQuery bind() event.

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("p").bind("click", function(){
8.         alert("This paragraph was clicked.");
9.     });
10. });
11. </script>
12. </head>
13. <body>
14. <p>Click on the statement.</p>
15. </body>
16. </html>
```

jQuery focus()

The jQuery focus event occurs when an element gains focus. It is generated by a mouse click or by navigating to it.

This event is implicitly used to limited sets of elements such as form elements like <input>, <select> etc. and links <a href>. The focused elements are usually highlighted in some way by the browsers.

The focus method is often used together with blur () method.

Example of jQuery focus() event

Let's take an example to demonstrate jQuery focus() event.

```
1. <!doctype html>
2. <html lang="en">
3. <head>
4.   <meta charset="utf-8">
5.   <title>focus demo</title>
6.   <style>
7.     span {
8.       display: none;
9.     }
10.  </style>
11.  <script src="https://code.jquery.com/jquery-1.10.2.js"></script>
12. </head>
13. <body>
14.  <p><input type="text"> <span>Focus starts.. Write your name.</span></p>
15.  <p><input type="password"> <span>Focus starts.. Write your password.</span></p>
16.  <script>
17.    $( "input" ).focus(function() {
18.      $( this ).next( "span" ).css( "display", "inline" ).fadeOut( 2000 );
19.    });
20.  </script>
21. </body>
22. </html>
```

jQuery change()

jQuery change event occurs when the value of an element is changed. It works only on form fields. When the change event occurs, the change () method attaches a function with it to run.

Note: This event is limited to <input> elements, <textarea> boxes and <select> elements.

- **For select boxes, checkboxes, and radio buttons:** The event is fired immediately when the user makes a selection with the mouse.
- **For the other element types:** The event is occurred when the field loses focus.

Syntax:

1. \$(selector).change()
It triggers the change event for selected elements.
1. \$(selector).change(function)

Example of jQuery change() event

Let's take an example to demonstrate jQuery change() event.

1. <!DOCTYPE html>
2. <html lang="en">
3. <head>
4. <meta charset="utf-8">
5. <title>change demo</title>
6. <style>
7. div {
8. color: red;
9. }
10. </style>
11. <script src="https://code.jquery.com/jquery-1.10.2.js"></script>
12. </head>
13. <body>
14. <select id="se" name="actors" >
15. <option>Uthappa</option>
16. <option selected="selected">Kattapa</option>
17. <option>Veerappa</option>
18. <option>Bahubali</option>
19. <option>Bhallal Dev</option>
20. <option>Awantika</option>
21. </select>
22. <div id="loc"></div>
23. <script>


```

24. $( "select" ) .change(function () {
25. document.getElementById("loc").innerHTML="You selected: "+document.getElementById("s
    e").value;
26. });
27. </script>
28. </body>
29. </html>

```

jQuery submit()

jQuery submit event is sent to the element when the user attempts to submit a form.

This event is only attached to the <form> element. Forms can be submitted either by clicking on the submit button or by pressing the enter button on the keyboard when that certain form elements have focus. When the submit event occurs, the submit() method attaches a function with it to run.

Syntax:

1. \$(selector).submit()
It triggers the submit event for selected elements.
1. \$(selector).submit(function)

Example of jQuery submit() event

Let's take an example to demonstrate jQuery submit() event.

```

1. <!DOCTYPE html>
2. <html lang="en">
3. <head>
4.   <meta charset="utf-8">
5.   <title>submit demo</title>
6.   <style>
7.     p {
8.       margin: 0;
9.       color: blue;
10.    }
11.    div,p {
12.      margin-left: 10px;
13.    }
14.    span {
15.      color: red;
16.    }

```

```

17. </style>
18. <script src="https://code.jquery.com/jquery-1.10.2.js"></script>
19. </head>
20. <body>
21. <p>Type 'javatpoint' to submit this form finally.</p>
22. <form action="javascript:alert( 'success!' );">
23. <div>
24.   <input type="text">
25.   <input type="submit">
26. </div>
27. </form>
28. <span></span>
29. <script>
30. $( "form" ).submit(function( event ) {
31.   if ( $( "input:first" ).val() === "javatpoint" ) {
32.     $( "span" ).text( "Submitted Successfully." ).show();
33.     return;
34.   }
35.   $( "span" ).text( "Not valid!" ).show().fadeOut( 2000 );
36.   event.preventDefault();
37. });
38. </script>
39. </body>
40. </html>

```

jQuery keydown()

When you press a key on the keyboard, the `keydown()` event is occurred and once the `keydown()` event is occurred, it executes the function associated with `keydown()` method to run.

The `keydown()` event is generally used with two other events.

- **Keypress() event:** It specifies that the key is pressed down.
- **KeyUp() event:** It specifies that the key is released.

Example of jQuery keydown() event

Let's take an example to demonstrate jQuery `keydown()` event.

```

1. <!DOCTYPE html>
2. <html>

```

3. **<head>**
4. **<script src="https://code.jquery.com/jquery-1.10.2.js"></script>**
5. **<script>**
6. `$(document).ready(function(){`
7. `$("input").keydown(function(){`
8. `$("input").css("background-color", "green");`
9. `});`
10. `$("input").keyup(function(){`
11. `$("input").css("background-color", "violet");`
12. `});`
13. `});`
14. **</script>**
15. **</head>**
16. **<body>**
17. Write something: **<input type="text">**
18. **</body>**
19. **</html>**

jQuery keypress()

The jQuery keypress () event is occurred when a keyboard button is pressed down. This event is similar to keydown() event. The keypress() method is executed or attach a function to run when a keypress() event occurs.

Example of jQuery keypress() event

Let's take an example to demonstrate jQuery keypress() event.

1. **<!DOCTYPE html>**
2. **<html>**
3. **<head>**
4. **<script src="https://code.jquery.com/jquery-1.10.2.js"></script>**
5. **<script>**
6. `i = 0;`
7. `$(document).ready(function(){`
8. `$("input").keypress(function(){`
9. `$("span").text (i += 1);`
10. `});`
11. `});`
12. **</script>**
13. **</head>**

14. **<body>**
15. Write something: **<input type="text">**
16. **<p>**Keypresses: **0</p>**
17. **</body>**
18. **</html>**

jQuery mouseenter()

The mouseenter() method adds an event handler function to an HTML element. This function is executed, when the mouse pointer enters the HTML element.

When you enter your mouse cursor over the selected element, it triggers the mouseenter event and once the mouseenter event is occurred, it executes the mouseenter() method to attach the event handler function to run.

This event is generally used together with mouseleave() event.

Syntax:

1. **\$(selector).mouseenter()**
It triggers the mouseenter event for selected elements.
1. **\$(selector).mouseenter(function)**

Example of jQuery mouseenter() event

Let's take an example to demonstrate jQuery mouseenter() event.

1. **<!DOCTYPE html>**
2. **<html>**
3. **<head>**
4. **<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>**
5. **<script>**
6. **\$(document).ready(function(){**
7. **\$("#h1").mouseenter(function(){**
8. **\$("div").text("Mouse entered on heading").show().fadeOut(2000);**
9. **});**
10. **});**
11. **</script>**
12. **</head>**
13. **<body>**
14. **<h1 id="h1">Enter this heading.</h1>**
15. **<div></div>**
16. **</body>**

17. `</html>`

jQuery mouseenter() event example 2

Let's see another example of jQuery mouseenter() event.

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("p").mouseenter(function(){
8.         $("p").css("background-color", "lightgreen");
9.     });
10.    $("p").mouseleave(function(){
11.        $("p").css("background-color", "yellow");
12.    });
13. });
14. </script>
15. </head>
16. <body>
17. <p>Move your mouse cursor over this statement.</p>
18. </body>
19. </html>
```

jQuery hover()

The jQuery hover() method executes two functions when you roam the mouse pointer over the selected element. The hover() method triggers both the mouseenter and mouseleave events.

Syntax:

```
1. $(selector).hover(inFunction,outFunction)
```

jQuery hover() example

Let's take an example to see the hover () effect. In this example, when you hover your mouse pointer over the selected element the the background color of that selected element will be changed.

```
1. <!DOCTYPE html>
2. <html>
```

```

3. <head>
4. <script src="https://code.jquery.com/jquery-1.10.2.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("p").hover(function(){
8.         $(this).css("background-color", "violet");
9.     }, function(){
10.        $(this).css("background-color", "green");
11.    });
12. });
13. </script>
14. </head>
15. <body>
16. <p>Hover your mouse pointer on me!</p>
17. </body>
18. </html>

```

jQuery mouseover()

The mouseover event is occurred when you put your mouse cursor over the selected element .Once the mouseover event is occurred, it executes the mouseover () method or attach a function to run.

This event is generally used with mouseout() event.

Note: Most of the people are confused between mouseenter and mouseover.

Difference between mouseenter() and mouseover()

The mouseenter event is only triggered if the mouse pointer enters the selected element whereas the mouseover event triggers if the mouse cursor enters any child elements as well as the selected element.

Syntax:

1. \$(selector).mouseover()
It triggers the mouseover event for selected elements.
1. \$(selector).mouseover(function)

Example of jQuery mouseover() event

Let's take an example to demonstrate jQuery mouseover() event.

1. <!DOCTYPE html>
2. <html>
3. <head>

```
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("p").mouseover(function(){
8.         $("p").css("background-color", "lightgreen");
9.     });
10.    $("p").mouseout(function(){
11.        $("p").css("background-color", "orange");
12.    });
13. });
14. </script>
15. </head>
16. <body>
17. <p>Move your cursor over this paragraph.</p>
18. </body>
19. </html>
```

jQuery load()

The load () method is used to load a specific element. It attaches an event handler to load event. It was deprecated in jQuery 1.8 version of jQuery library.

The load event occurs when a specific element is loaded. It is generally used with a URL (image, script, frame, iframe), and the window object.

Note: On some browsers, the load event did not trigger if the image is cached.

Syntax:

```
1. $(selector).load(function)
```

Example of jQuery load() event

Let's take an example to demonstrate jQuery load() event.

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
7.     $("img").load(function(){
```

```
8.     alert("Image loaded.");
9.   });
10. });
11. </script>
12. </head>
13. <body>
14. 
15. <p><b>Note:</b> On some browsers, the load event did not trigger if the image is cache
    d.</p>
16. </body>
17. </html>
```

jQuery unload()

The jQuery unload() method is used to unload a specific element. It attaches an event handler to unload event. The unload event is sent to the window element when the user navigates away from the page. It was deprecated in jQuery 1.8 version of jQuery library.

Ways to trigger unload event

An unload event is triggered if you:

- Click on a link which leads to leave the page.
- Use the forward or back button.
- Type a new URL in the address bar.
- Close the browser window.
- Reload the page.

Syntax:

```
1. $(selector).unload(function)
```

Example of jQuery unload() event

Let's take an example to demonstrate jQuery unload() event.

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
5. <script>
6. $(document).ready(function(){
```



```
7.    $(window).unload(function(){
8.        alert("Goodbye!");
9.    });
10. });
11. </script>
12. </head>
13. <body>
14. <p>When you click <a href="http://www.javatpoint.com/">this link</a>, or close the win
    dow,
15. unload event will be triggered.</p>
16. </body>
17. </html>
```