



jQuery

Mr. Devang Bhagdev

NodeJS Lead – WebCodeGenie Technology Pvt. Ltd

Mr. Bhavesh Parmar

Jr. Software Developer – WebCodeGenie Technology Pvt. Ltd



What is jQuery?

- jQuery is a JavaScript library designed to simplify HTML DOM tree traversal and manipulation, as well as event handling, CSS animation, and Ajax.
- It is free and open-source .
- jQuery was originally created in January 2006 at BarCamp NYC by John Resig
- jQuery means "write less do more".
- jQuery simplifies AJAX call and DOM manipulation.

Add jQuery

```
<head>  
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.3/jquery.min.js"></script>  
</head>
```



jQuery Syntax



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)
- The factory function `$()` is a synonym of `jQuery()` function. So in case you are using any other JavaScript library where \$ sign is conflicting with something else then you can replace \$ sign by jQuery name and you can use function `jQuery()` instead of `$()`.

\$(document).ready()



- The code inserted between `$(document).ready()` is executed only once when page is ready for JavaScript code to execute.

```
$(document).ready(function() {  
    $("p").css("color", "red");  
});
```

- In place of `$(document).ready()`, you can use shorthand notation `$(function() {` only.

```
$(function() {  
    $("p").css("color", "red");  
});
```



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 name, 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.

1. The element Selector

`$("p")` - select all `<p>` elements on a page like this:

2. The #id Selector

`$("#test")` - select first element who have id "test"

3. The .class Selector

`$(".test")` - select all elements with class "test".



Some Important Selectors

- 1. **\$(*)** - Universal Selector, Selects all elements
- 1. **\$(this)** - Selects the current HTML element
- 1. **\$("p.intro")** - Selects all `<p>` elements with `class="intro"`
- 1. **\$("p:first")** - Selects the first `<p>` element
- 1. **[attribute]** - **\$("[href]")** - Selects all elements with an href attribute
- 1. **:checked** **\$(":checked")** - All checked input elements
- 1. **\$(".intro,.demo")** - All elements with the class "intro" or "demo"
- 1. **parent > child** - **\$("div > p")** - All `<p>` elements that are a direct child of a `<div>` element
- 1. **parent descendant** - **\$("div p")** - All `<p>` elements that are descendants of a `<div>` element
- 1. **element + next** - **\$("div + p")** - The `<p>` element that are next to each `<div>` elements
- 1. **element ~ siblings** - **\$("div ~ p")** - All `<p>` elements that appear after the `<div>` element

jQuery Event Methods



Mouse Events

click

dblclick

mouseenter

mouseleave

Keyboard Events

keypress

keydown

keyup

Form Events

submit

change

focus

blur

Document/Window Events

load

resize

scroll

unload

Syntax:

```
$("#p").on("click", function(){  
    // action goes here!!  
});
```

```
$("#p").on("click", function(){  
    alert("The paragraph was clicked.");  
});
```



jQuery Effects



1. `hide/show`
2. `slideUp/slideDown/slideToggle`
3. `fadeIn/fadeOut/fadeToggle/fadeTo`

`$(selector).hide(speed,callback);`

- The optional speed parameter specifies the speed of the hiding/showing, and can take the following values: "slow", "fast", or milliseconds.
 - The optional callback parameter is a function to be executed after the hide() or show() method completes
4. **animate** - The jQuery animate() method is used to create custom animations.
`$(selector).animate({params},speed,callback);`
5. **stop** - `$(selector).stop(stopAll,goToEnd);`
6. **jQuery Method Chaining** - Chaining allows us to run multiple jQuery methods (on the same element) within a single statement.
e.g, `$("#p1").slideUp(2000).slideDown(2000);`

jQuery - Get And Set Content



1. Get Contents

Three simple, but useful, jQuery methods for DOM manipulation are:

text() - Sets or returns the text content of selected elements - `$("#test").text()`

html() - Sets or returns the content of selected elements (including HTML markup) - `$("#test").html()`

val() - Sets or returns the value of form fields - `$("#test").val()`

2. Set Content

We will use the same three methods from the previous page to set content:

text() - Sets or returns the text content of selected elements - `text("Hello world!");`

html() - Sets or returns the content of selected elements (including HTML markup) - `$("#test2").html("Hello world!");`

val() - Sets or returns the value of form fields - `$("#test3").val("Namste Bharat");`

jQuery - Get And Set Attributes - attr()



1. Get Attribute

The jQuery attr() method is used to get attribute values.

e.g., `$("#test").attr("href")`

1. Get Attribute

The jQuery attr() method is also used to set/change attribute values.

e.g., `$("#test").attr("href", "https://webcodegenie.com/");`

```
$("#test").attr({  
  "href" : "https://www.w3schools.com/jquery/",  
  "title" : "W3Schools jQuery Tutorial"  
});
```

jQuery - Get And Set Data Attribute



1. Attach Data to an Element

```
$(selector).data(name,value);
```

2. Return Data from an Element

```
$(selector).data(name);
```

jQuery Manipulating CSS



- **addClass()** - Adds one or more classes to the selected elements

```
$("h1, h2, p").addClass("test");
```

- **removeClass()** - Removes one or more classes from the selected elements

```
$("h1, h2, p").removeClass("test");
```

- **toggleClass()** - Toggles between adding/removing classes from the selected elements

```
$("h1, h2, p").toggleClass("test");
```

jQuery - css() Method



Return a CSS Property

To return the value of a specified CSS property, use the following syntax:

```
css("propertyname");    $("#p").css("background-color");
```

Set a CSS Property

To set a specified CSS property, use the following syntax:

```
css("propertyname","value");    $("#p").css("background-color", "yellow");
```

```
css({"propertyname":"value","propertyname":"value",...});
```

「 Thank You 」

Get in Touch With Us

Devang Bhagdev (devang@webcodegenie.com)
Bhavesh Parmar (bhavesh.p@webcodegenie.com)