

Software engineering

jQuery Tutorial 2021.

reference: w3shools & tutorialspoint

1. What is jQuery?

If you want to study jQuery subjects, you should have a basic knowledge of HTML, CSS, JavaScript. jQuery is a light weight, "write less, do more", JavaScript library. 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.

Example)

```
<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/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</button>
</body>
</html>
```

Practice #1) run and capture the above code.

Javascript VS jQuery)

```
<!DOCTYPE html>
<html>
<!-- javascript -->
<head>
  <link rel="stylesheet" href="style.css">
  <script src="script.js"> </script>
  <script>
    function sayHello(){
      alert('안녕!');
    }

  </script>
</head>

<body onload="sayHello()">
  <h1>Hello javascript!</h1>
</body>

</html>
```

```
<!DOCTYPE html>
<html>
<!-- jQuery -->
<head>
  <script src="http://code.jquery.com/jquery-2.1.3.min.js"> </script>
  <link rel="stylesheet" href="style.css" />
  <script>
    $(document).ready(function(){
      alert('안녕!');
    });
  </script>
</head>

<body>
  <h1>Hello jQuery!</h1>
</body>

</html>
```

jQuery & CSS)

```
<html>

  <head>
    <title>The jQuery Example</title>
    <script type = "text/javascript"
      src
"https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js"> </script>

    <script type = "text/javascript" language = "javascript">
      $(document).ready(function() {
        $("div:first").width(100);
        $("div:first").css("background-color", "black");
      });
    </script>

    <style>
      div{ width:70px; height:50px; float:left; margin:5px; background:yellow;
cursor:pointer; }
    </style>
  </head>

  <body>

    <div>1.</div>
    <div>2.</div>
    <div>3.</div>
    <div>4.</div>
    <div>5.</div>

  </body>

</html>
```

2. selectors

The jQuery library harnesses the power of Cascading Style Sheets (CSS) selectors to let us quickly and easily access elements or groups of elements in the Document Object Model (DOM).

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.

Example selectors)

```
<html>

  <head>
    <title>The jQuery Example</title>
    <script type = "text/javascript"
      src
"https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js"> </script>

    <script type = "text/javascript" language = "javascript">
      $(document).ready(function() {
        $("p").css("background-color", "yellow");
      });
    </script>
  </head>

  <body>
    <div>
      <p class = "myclass">This is a paragraph.</p>
      <p id = "myid">This is second paragraph.</p>
      <p>This is third paragraph.</p>
    </div>
  </body>

</html>
```

Practice #2) Run the above code and capture it.

The \$() factory function

jQuery selectors start with the dollar sign and parentheses – `$()`. The factory function `$()` makes use of following three building blocks while selecting elements in a given document –

S.N	Selector & Description
1. Tag Name	Represents a tag name available in the DOM. For example <code>\$('p')</code> selects all paragraphs <code><p></code> in the document.
2. Tag ID	Represents a tag available with the given ID in the DOM. For example <code>\$('#some-id')</code> selects the single element in the document that has an ID of some-id.
3. Tag Class	Represents a tag available with the given class in the DOM. For example <code>\$('.some-class')</code> selects all elements in the document that have a class of some-class.

All the above items can be used either on their own or in combination with other selectors. All the jQuery selectors are based on the same principle except some tweaking.

Practice #3) modify `$("p").css` to `$(".myclass").css` to change background color of the first paragraph. Run and capture the result.

Practice #4) modify `$("p").css` to `$("#myid").css` to change background color of the second paragraph. Run and capture the result.

3. Event

All the different visitor's actions that a web page can respond to are called events. An event represents the precise moment when something happens. Such as moving a mouse over an element, selecting a radio button or clicking on an element

Mouse Events	Keyboard Events	Form Events	Document/ Window Events
Click	keypress	Submit	Load
Dblclick	Keydown	Change	Resize

mouseenter	Keyup	Focus	Scroll
mouseleave		blur	Unload

Example event)

```

<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"> </script>
<script>
$(document).ready(function(){
    $("p").on({
        mouseenter: function(){
            $(this).css("background-color", "lightgray");
        },
        mouseleave: function(){
            $(this).css("background-color", "lightblue");
        },
        click: function(){
            $(this).css("background-color", "yellow");
        }
    });
});
</script>
</head>
<body>

<p>Click or move the mouse pointer over this paragraph.</p>

</body>
</html>

```

Practice #5) Run the above code and capture the result.

4. Effect_fadeIn()

There are a lot of effects we can choose. Such as hide/show, fade, slide, animate, stop, callback and chaining. First, let's look at fadeIn() Method.

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

```

<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/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>

```

5. Effect_slideToggle()

Next, slideDown() and slideUp() method. 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. However, if you want both effect you are able to use slideToggle(). 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. Here is example of it.

Example slideToggle()


```

<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/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: #e5eecd;
    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>

```

6. Effect_animate()

The jQuery animate() method is used to create custom animations. By default, jQuery comes with queue functionality for animations. This means that if you write multiple animate() calls after each other, jQuery creates an "internal" queue with these method calls. Then it runs the animate calls ONE by ONE. So, if you want to perform different animations after each other, we take advantage of the queue functionality:

Example animate()

```
<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
<script>
$(document).ready(function(){
    $("button").click(function(){
        var div = $("div");
        div.animate({height: '300px', opacity: '0.4'}, "slow");
        div.animate({width: '300px', opacity: '0.8'}, "slow");
        div.animate({height: '100px', opacity: '0.4'}, "slow");
        div.animate({width: '100px', opacity: '0.8'}, "slow");
    });
});
</script>
</head>
<body>

<button>Start Animation</button>

<p>By default, all HTML elements have a static position, and cannot be moved. To
manipulate the position, remember to first set the CSS position property of the element to
relative, fixed, or absolute!</p>

<div style="background:#98bf21;height:100px;width:100px;position:absolute;"></div>

</body>
</html>
```

Practice #6) Run the above code and capture the result.

7. Effect_chaining()

Until now we have been writing jQuery statements one at a time (one after the other). However, there is a technique called chaining, that allows us to run multiple jQuery commands, one after

the other, on the same element(s). Tip: This way, browsers do not have to find the same element(s) more than once. To chain an action, you simply append the action to the previous action. When chaining, the line of code could become quite long. However, jQuery is not very strict on the syntax; you can format it like you want, including line breaks and indentations.

```
<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("#p1").css("color", "red")
        .slideUp(2000)
        .slideDown(2000);
    });
});
</script>
</head>
<body>

<p id="p1">jQuery is fun!!</p>

<button>Click me</button>

</body>
</html>
```

Practice #7) Run the above code and capture the result.