jQuery
quick view

GOCODE

What is jQuery?

jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers.

With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript.

Getting started

HTML:

```html
<button>CLICK Here 1</button>
<button>CLICK Here 2</button>
```

JS:

```html
<script src="https://unpkg.com/jquery"></script>
<script>
$(document).ready(function () {
    $("button").click(function (event) {
      alert("Thanks for clicking!");
    });
});
</script>
```

Use minified version for production:
https://unpkg.com/jquery@3.3.1/dist/jquery.min.js

In vanilla JS:

```js
document.addEventListener('DOMContentLoaded',
function() {
    // DOM ready, run it!
}, false);
```

Or, Just put script on the bottom of body!
https://stackoverflow.com/questions/6026645/document-readyfunction-vs-script-at-the-bottom-of-page
https://css-tricks.com/lodge/learn-jquery/05-dom-ready/

$(document.ready).function() { CODE }   or $(function() { CODE })
Use this event to run code as soon as the document is ready to be manipulated so can put this code everywhere you want. Like window.onload but doesn't wait the images to load. **Put all your jQuery code here.**

$
jQuery Selector. Like document.querySelectorAll but with no need in loop to run addEventListener separately.

jQuery Cheat sheets

https://oscarotero.com/jquery/

https://htmlcheatsheet.com/jquery/

http://overapi.com/jquery

Selectors

Some are the same as document.querySelector and CSS selectors...

```
$( "#myId" );    $( ".myClass" );    $( "input[name='first_name']" );    $( "#contents ul.people li" );
```

But some are more powerful...

```
Selects all buttons (button elements and input type="button" elements)

$( ":button" );

Selects all <input>, <textarea>, <select>, and <button> elements

$( "form :input" );
```

More: https://learn.jquery.com/using-jquery-core/selecting-elements/#selecting-by-type

Check if my query contains elements

```
if ( $( "div.foo" ).length ) {
    ...
}
```

Selectors

Filtering selections

```
// Refining selections.
 $( "div.foo" ).has( "p" );          // div.foo elements that contain <p> tags
 $( "h1" ).not( ".bar" );            // h1 elements that don't have a class of bar
 $( "ul li" ).filter( ".current" ); // unordered list items with class of current
 $( "ul li" ).first();               // just the first unordered list item
 $( "ul li" ).eq( 5 );               // the sixth
```

Manipulation

Add class

```
// jQuery
$('div').addClass('myClass');

// JavaScript
var div = document.querySelector('div');
div.classList.add('myClass');
```

Remove class

```
// jQuery
$('div').removeClass('myClass');

// JavaScript
var div = document.querySelector('div');
div.classList.remove('myClass');
```

Toggle class

```
// jQuery
$('div').toggleClass('myClass');

// JavaScript
var div = document.querySelector('div');
div.classList.toggle('myClass');
```

Manipulation for all elements

```
// jQuery
var someElem = $('.someElem');
someElem.addClass('myClass');

// JavaScript - this adds the class to the first Node only!
var someElem = document.querySelector('.someElem');
someElem.classList.add('myClass');

// JavaScript - this adds the class to every Node in the NodeList
var someElem = document.querySelectorAll('.someElem');
for (var i = 0; i < someElem.length; i++) {
  someElem[i].classList.add('myClass');
}
```

Set Content with html(htmlString)

```
<button>Change content of all p elements</button>

<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("p").html("Hello <b>world!</b>");
  });
});
</script>
```

```
like:
document.querySelector('p').innerHTML = "Hello <b>world!</b>"
```

You can also use text() to change text...

show(), hide(), toggle()

```
<button onclick="$('div').show()">Check Show</button>
<button onclick="$('div').hide()">Check Hide</button>
<button onclick="$('div').toggle();">Check Toggle</button>
<div class="check" style="display: none">MY TEXT</div>
```

show with animation - show(duration, callback)

```
<div id="clickme">
  Click here
</div>
<img
  id="book"
  style="display: none"
  src="https://image.freepik.com/free-vector/books-stack-realistic_1284-4735.jpg"
  alt=""
  width="100"
  height="123"
/>
<script>
  $('#clickme').click(function() {
    $('#book').show('slow', function() {
      alert('ANIMATION COMPLETE!');
    });
  });
</script>
```

Events

Single event

```
$(elem).on('click', function () {
  // ...
});
```

Or:

```
$(elem).click(function () {
  // ...
});
```

Chained Events

```
$(elem).on('click focus keyup', function () {
  // ...
});
```

$(this)

$(this) represent the same element that we working on right now…

Example - hide the button that you clicked on it

```
$("button").click(function (event) {
    $(this).hide();
});
```

CSS Manipulation

```
// jQuery
$(elem).css({
  "background" : "#F60",
  "color" : "#FFF"
});

// JavaScript
var elem = document.querySelector(elem);
elem.style.background = '#F60';
elem.style.color = '#FFF';
```

jQuery AJAX

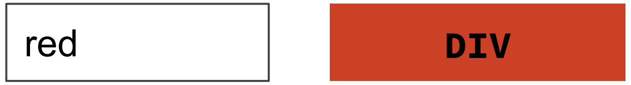
```
// Using the core $.ajax() method
$.ajax({
    // The URL for the request
    url: "post.php",
    // The data to send
    // (will be converted to a query string)
    data: {
        id: 123
    },
    // Whether this is a POST or GET request
    type: "GET",

    // The type of data we expect back
    dataType : "json",
})
```

```
// Code to run if the request succeeds (is done);
// The response is passed to the function
.done(function( json ) {
   $( "<h1>" ).text( json.title ).appendTo( "body" );
   $( "<div class=\"content\">").html( json.html ).appendTo( "body" );
})
// Code to run if the request fails; the raw request and
// status codes are passed to the function
.fail(function( xhr, status, errorThrown ) {
  alert( "Sorry, there was a problem!" );
  console.log( "Error: " + errorThrown );
  console.log( "Status: " + status );
  console.dir( xhr );
})
// Code to run regardless of success or failure;
.always(function( xhr, status ) {
  alert( "The request is complete!" );
});
```

jQuery - Exercises

Exercise 1 - Getting started

Use jQuery in your page with a text input and div and change the background color of the div according to the input value. For example:



Exercise 2 - Your jQuery Countdown

Convert your countdown exercise into jQuery

Exercise 3 - Your jQuery Form

Convert your register form exercise into jQuery

More Info

1. https://toddmotto.com/is-it-time-to-drop-jquery-essentials-to-learning-javascript-from-a-jquery-background/#css-manipulation
2.