**How to manuipulate classes using the classList API**

you don't need jQuery

I was doing some work recently and needed to change some CSS rules based on the classes in the HTML, which is indeed a very common thing to do.

In the past, I wholly relied on jQuery to handle DOM manipulations for me. However, in this case, it didn’t make sense to import the whole jQuery library just to perform this basic DOM manipulation. There had to be a better way, and indeed there was.

A quick search presented me with a variety of options to achieve what I was after but I went with the classList method because it’s easy to understand and browser support was good.

If you’re used to using jQuery exlusively for DOM manipulation, this is a great place to start gaining some independence over jQuery and you might be surprised at how easy it is sometimes.

**What is the classList API**

The HTML5 classList API provides a way to grab all the classes associated with an element so that you can use JavaScript to modify it.

Using the classList DOM property on an element will return a [DOMTokenList](https://developer.mozilla.org/en/docs/Web/API/DOMTokenList) which contains all the classes applied to an element and length property which signifies the total number of classes on that element.

Let’s take a look at this example:

<nav class="nav navbar navbar-nav" id="nav"></nav>

var nav = document.getElementById("nav");

console.log(nav.classList);

//returns

{

0: "nav"

1: "navbar"

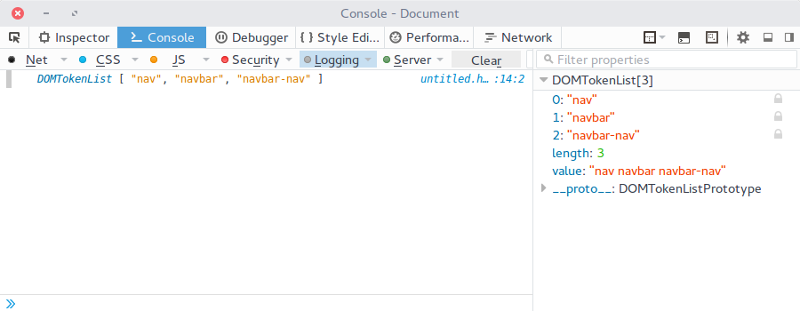
2: "navbar-nav"

length: 3

value: "nav navbar navbar-nav"

}

You can try the above in your browser to see it in action.



Getting the classes of an element is all well and good but it isn’t all that useful on its own. We need a way to manage and update those classes and the classList property provides a few methods that do just that:

* add(): Adds specified classes
* remove(): Removes specified classes
* contains(): Checks if specified class exists on the element
* toggle(): toggles specified class
* index(): returns class at a specified position in the list
* length: returns the number of classes

Let’s take a look at each one in turn.

**Adding classes**

Adding a class to an element is straightforward. Just apply the class name as an argument to the add() method. Note that if the class name already exists on the element, it won’t be added again.

<span class="heading" id="title"></span>

document.getElementById("span").classList.add("title");

//gives class="heading title"

To add multiple classes, seperate each class with a comma.

document.getElementById("span").classList.add("title", "headline");

//gives class="heading title headline"

**Removing classes**

To remove a class, all you need to do is pass the class name as an argument to the remove() method. If the class name does not already exist in the classList, an error is thrown.

<header class="masthead clearfix" id="header"></header>

document.getElementById("header").classList.remove("masthead");

//gives class="clearfix"

Removing multiple classes at once works in the same way as adding multiple classes.

**Check if a class exists**

Using the contains() method, we can check if a specified class is present in an element’s classList and perform operations based on the return value.

For example,

<button class="hidden" id="btn">Click Me</button>

var button = document.getElementById("btn");

if (button.classList.contains("hidden")) {

//do something

} else {

//do something else

}

**Toggling Classes**

Adding or removing a class based on user action is a very common thing to do and this was exactly what I wanted to achieve with classList.

Here’s what I eventually did:

<div class="menu" id="menu" onclick="hasClass()"></div>

var page = document.getElementById("page");

var menu = document.getElementById("menu");

var nav = document.getElementById("navigation");

function hasClass() {

page.classList.toggle("open");

menu.classList.toggle("active");

nav.classList.toggle("hidden");

}

**Check number of classes**

To find out how many classes are applied to an element, use the length property on the element.

<nav class="nav navbar navbar-nav" id="navbar"></nav>

document.getElementById("navbar").classList.length;

//returns 3

**Browser Support**

The classList API is supported by all mordern browsers including IE 10 and later. If you need to support older IE versions, you may need to find an alternative method or use a [polyfill](https://github.com/eligrey/classList.js).

**Wrap Up**

As you can see, the classList API is really easy to use so I encourage you to begin exploring its capabilities in your own applications. Do leave a comment if you have any questions, or reach out to me on [Twitter](https://twitter.com/ayisaiah).

**Further Reading**

* <http://davidwalsh.name/classlist>
* [Mozilla Developer Network: Element.classList](https://developer.mozilla.org/en-US/docs/Web/API/Element/classList)