**Keeping Time with Moment.js**

A clock with a white circle on it

Description automatically generated

**Moment.js**

In this tutorial we are going to look at Moment.js! As you can see from the subtitle in the picture above the Moment.js library will allow us to: "*Parse, validate, manipulate, and display times in JavaScript"*! [Moment.js](https://momentjs.com/) is an awesome JavaScript library that helps you manage dates, in the browser and in Node.js as well.

So let's get started. Like most tutorials we'll get you started coding, look at the results and break down the code together. This is what we are going to build:

A cartoon of a santa claus

Description automatically generated

Because Moment.js is a library we can't just open the Chrome Dev Tools Console and expect to have access to that library. The quickest way to have access to that library is to go to a specific page that is running the Moment.js library!

Where might that be? You guessed it, go to momentjs.com and open up the console.

A screenshot of a computer

Description automatically generated

While you are there type in: moment()

This will result in an output:

1. Moment {\_isAMomentObject: true, \_isUTC: false, \_pf: {…}, \_locale: Locale, \_d: Tue Jan 15 2019 01:04:30 GMT-0800 (Pacific Standard Time), …}

You will get the moment object full of properties that you can expand. You could get lost in those details so let's start with some moment methods that can give us really helpful information.

**Manipulating date and time**

The moment object gives us the ability to manipulate the date object however we choose, applying several (chaining) moment methods to it returns desired formatted date values.

**Date formatting**

Previously, formatting dates with the Date() string involved performing several string operations on the returned date string. With moment we just need to add the format() method to format the date. This method also receives parameters defining how we would like the date to appear. Try the following in your console (Make sure you are on the momentjs.com page when you do).

1. var today = moment().format() //returns 2019-01-15T21:07:52+01:00
2. var today = moment().format('YYYY-MM-DD') //returns 2019-01-15
3. var today = moment().format('YY/MMM/DD') // returns 19/Jan/15
4. var today = moment().format('D/MMMM/Y') // returns 15/January/19
5. var time = moment().format('ddd DD/MM/YYYY') //returns Wed 01/15/2019

**isBefore()**

As its name implies, the isBefore() method verifies if one date value is before another date value.

1. var then = moment("27/08/2019", "DD/MM/YYYY")
2. var now = moment("27/09/2018", "DD/MM/YYYY")
3. then.isBefore(now) // false
4. now.isBefore(then) // true

**subtract()**

This method subtracts time from the original moment. Here, the subtract()method is called and passed parameters of a number denoting the amount of time to be removed and the time value. This time value can also be written in shorthand, they include:

* d: Days
* y: Years
* Q: Quarters
* M: Months
* w: Weeks
* h: Hours
* m: Minutes
* s: Seconds
* Ms: Milliseconds

1. var lastWeek = moment().subtract(7, 'days')
2. //subtracts 7 days from the current date

**add()**

This is similar to the subtract() method and only adds a desired amount of time to the moment.

1. var nextHour = moment().add(1, 'hour')
2. // adds 1 hour from the current date

**diff()**

This is one of the common problems encountered when manipulating time and date. With moment, the diff() method is used to calculate the difference between two moments and it returns the difference in milliseconds. However, other units of time such as years, months, weeks, days, hours, minutes, and seconds can be returned. The diff() method takes in a unit of time as a first parameter and can take the desired time unit as a second parameter.

1. var time = moment("27/08/2019", "DD/MM/YYYY")
2. var time2 = moment("27/09/2019", "DD/MM/YYYY")
3. console.log(time.diff(time2,"days") + ' days') // 334days

Ok, that's all we need! Let's build something!

**Let's Build Something!**

A cartoon of a santa claus

Description automatically generated

It's a Christmas Countdown! This is always a practical countdown to vacation, family gatherings and budgeting for gifts!

We have all the tools we need with Moment.js methods to make this happen.

So let's build this together then walk through it!

First, let's build out the structure of the page with HTML. In the extra resources for this tutorial I walk through the text line by line.

HTML:

1. <body>
2. <!-- image tag with src attribute pointing to hosted Santa image -->
3. <img src="https://github.com/robgmerrill/online-course-images/blob/master/001-santa-claus.png?raw=true"/>
4. <h1>Count days till next christmas</h1>
5. <!-- h2 tag with an id of countdown that we will populate with moment library -->
6. <h2 id="countdown"></h2>
7. </body>
8. <!-- This allows you to have access to the moment library -->
9. <script src="https://unpkg.com/moment" />

The result should look like this:

A cartoon of a santa claus

Description automatically generated

Currently, our image is aligned left instead of center and there is no gradient background. We are also missing the day(s) remaining until Christmas. Now, let's add the CSS:

1. body {
2. /\* fallback background is browser doesn't support linear gradients \*/
3. background-color: #89d8d3;
4. /\* gradient image \*/
5. background-image: linear-gradient(315deg, #89d8d3 0%, #03c8a8 74%);
6. /\* 100% of the height \*/
7. height: 100vh;
8. /\* set the body display to grid \*/
9. display: grid;
10. /\* center the grid contents \*/
11. place-items: center;
12. }

The result of your work should be the following:

A cartoon of a santa claus

Description automatically generated

Ok, now we are matching our original design except for the JavaScript that will display the number of day(s) till Christmas. Here is the code. Not to be redundant but I talk through the code in the tutorial.

JavaScript:

1. // declare variable `now` and set to the current time
2. const now = moment();
3. // declare variable `christmas` and set to December 25th
4. let christmas = moment("Dec 25", "MMM DD");
5. // create a conditional to check if `christmas` is before `now` by passing in `now` and `days` as parameters into the `isBefore()` method
6. if ( christmas.isBefore(now, 'days') ) {
7. // if `christmas` is before the current date then add a year to the value of `christmas`
8. christmas = christmas.add(1, 'year');
9. }
10. // declare variable `diff` and set it equal to the difference between `christmas` and `now`
11. let diff = christmas.diff(now, 'days');
12. // declare variable `counter` and use a DOM method to target the `h2` tag with id of `countdown`
13. let counter = document.getElementById('countdown');
14. // set the `counter` variable `.innerHTML` property to the value of `diff` to display days till Christmas
15. counter.innerHTML = diff;

And that is it!! Our Christmas countdown is complete!

A cartoon of a santa claus

Description automatically generated