

Pontus Tahir
Handledare David Jensen
Javascript 2
24 februari 2020

Ramverk Moment.js och Moment Timezone

"Vad är klockan? och var?"



Moment.js 2.24.0

Parse, validate, manipulate, and display dates and times in JavaScript.



Moment Timezone 0.5.28-2019c

Parse and display dates in any timezone.

Inledning

Moment.js är ett av de äldsta och mest kända datum-relaterade biblioteken för både Node.js och JavaScript. Och tack vare detta så har de haft god tid på sig att göra det kompatibelt med alla de vanligaste webbutvecklingsverktygen, som till exempel Bower, require.js, Browsify, Typescript, Meteor och många fler. Det erbjuder ett jättebra API utan att behöva modifiera datumobjektets ursprungliga mall(logrocket.com). Istället skapas ett "wrapper"-object runt omkring det, för att försäkra sig om att inte röra till det i det globala scopet om jag förstår det hela rätt.

Moment är skapat för att fungera både i browsern och i Node.js. All kod ska gå att använda i dessa två miljöer, och alla tester görs också i båda miljöerna. Man kan alltså testa koden i sin browsers terminal direkt om man vill.(momentjs.com/docs).

Jag använde mig av cdn-länkar i min html när jag gjorde mitt testprojekt och det fungerade direkt helt utan krångel. Syntaxen man använder sig av är superenkel och finns väldigt väl dokumenterad på Moment.js egen hemsida. Så här ser länkarna ut i min HTML fil:

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/moment.js/2.18.1/moment.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/moment-timezone/0.5.27/moment-timezone-with-data.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/moment-timezone/0.5.27/moment-timezone-utils.js"></script>
```

En för Moment och två för Timezone(för att kunna hitta andra tidszoner).
(momentjs.com/docs).

Bakgrund

Tim Wood är namnet på skaparen av denna datum-formatterare. Från början inspirerad underscore.js och dess enkelhet, så "slängde" han 2011 ihop en plugin som han kallade för underscore.date till en början. Det utvecklades senare till Moment.js. Med över 43 000 stjärnor och 487 "contributors" på GitHub så är detta det mest använda biblioteket för datumhantering i javascript. Efter några år så gick Tim in i väggen pga sjukdom i familjen och sedan även personliga problem med depression. Han ägnade sig till slut mest åt Timezone istället med även det backade han ifrån och 2016 lämnade han över allt i trygga händer hos kollegan Iskren Chernev.

Han har efter det gjort bland annat en drink mixing app som heter Lush och arbetar nu som Senior Front-end Developer på RED Interactive Agency.(@timrwood/moment-end-of-term, 2016). (timwood.work).

Användning

Installation via npm om man inte vill hålla på med cdn-länkar är lätt och smidigt och Moment är fullt kompatibelt med React. Kör bara följande kommandon i din terminal:

```
npm install moment
```

för att installera moment.

Med react:

```
npm install --save moment react-moment
```

Och vill man använda typescript så importerar du bara enligt följande:

```
import * as moment from 'moment';
```

(npmjs.com)

För att ge ett bra exempel på användning utöver det vanliga, som att hämta datum till en kalender eller liknande, så finns en massa användbara plugins att tillgå. Vi kan vi till exempel titta på en vanlig sociala-medie-app där man kan se hur länge sedan ett inlägg publicerades. Där kan vi använda oss av ett format som tillåter oss att skriva ut datum och tid så här: 1h för “1 hour ago” eller 2d för “2 days ago”).

Koden skulle se ut ungefär som följer:

```
moment().subtract(2, 'days').twitterShort();
```

Returnerar "2d".(logrocket.com)

Den stora fördelen framför att hålla på med JavaScripts egna Date()-funktion är den minimala och tydliga syntaxen. speciellt för en rookie som jag. När jag gjorde mitt kod-exempel så var det bara att titta på Moments egna dokument och plocka den koden jag behövde för att printa ut i önskat format.

```
let rightNow = moment().tz('Australia/Sydney').format('YYYY Do YYYY, h:mm:ss a'); //DET ÄR HÄR DET HÄNDER!!
```

(momentjs.com/timezone)

moment() för att kalla på biblioteket, .tz() för timezone och .format() för formatet helt enkelt. Som synes så skriver jag bara 'Land/Stad' för att hämta tid från angiven plats. De har listat alla platser som går att hämta och finns inte staden så kan man i alla fall hämta rätt tidszon. Alla platser finns dokumenterade på GitHub om man är nyfiken. drygt 800 platser finns tillgängliga.

(<https://github.com/moment/moment-timezone/blob/develop/data/packed/latest.json>)

[illegible]

kritik

Den eventuella baksidan kan ju vara att man ofta inte behöver tillgå mer än en funktion som har med tid och datum att göra. Moment.js kommer med ett jättestort bibliotek med ganska tung information om tidszoner och sådant. Därför kanske man lika gärna kan använda ett lite mindre och lättviktigare bibliotek i sin app eller på hemsidan man bygger. Just det tycks vara den allmänna kritiken när man kollar runt. Det kan också vara betydligt långsammare än andra i vissa situationer. Det verkar vara bäst i hämtning av "is same day" och "is before", men riktigt dåligt när det gäller parsing. Här är en test som visar på prestandan och hur Moment står sig bland andra alternativ:



Att många ändå använder Moment är nog för att kompatibiliteten och användarvänligheten är så god. Många nämner date-fns som ett säkrare och stabilare alternativ men att Timezone fortfarande är helt utan konkurrens. (<https://inventi.studio/en/blog/why-you-shouldnt-use-moment-js>)

Framtid

Så länge det inte finns något vettigt alternativ till Timezone så kommer nog den delen att fortsätta blomstra i alla fall. I en Jämförelse av 7 befintliga bibliotek på Front-end bloggen LogRocket så är slutsatsen att Moment.js fortfarande är en vinnare i sammanhanget men bör se upp för nykomlingen DayJS som använder sig av samma API som Moment. (logrocket.com). Den som lever får se helt enkelt.

```
26 |  
27 | moment('om hundra år').format('kan inte våren komma snart');
```

Referenslista:

<https://blog.logrocket.com/javascript-date-libraries/>

<https://momentjs.com/docs/>

<https://medium.com/@timrwood/moment-end-of-term-522d8965689>

<https://timwood.work/>

<https://www.npmjs.com/package/react-moment>

<https://blog.logrocket.com/javascript-date-libraries/>

<https://inventi.studio/en/blog/why-you-shouldnt-use-moment-js>

<https://github.com/moment/moment-timezone/blob/develop/data/packed/latest.json>