Introduction to Temporal

Steve Harrison
https://www.linkedin.com/in/stevenanthonyharrison/

6 October 2022

Temporal is a new ECMAScript spec that provides a native modern date/time API, including timezone support.

What’s wrong with Date?

- No support for time zones other than the user’s local time and UTC
- Parser behaviour so unreliable it is unusable
- Date object is mutable
- DST behaviour is unpredictable
- Computation APIs are unwieldy
- No support for non-Gregorian calendars

Examples

https://github.com/steveharrison/introduction-to-temporal

Polyfills

@js-temporal/polyfill

https://github.com/js-temporal/temporal-polyfill

54.1kB minified + gzipped

```
import { Temporal, Intl, toTemporalInstant } from '@js-temporal/polyfill';
Date.prototype.toTemporalInstant = toTemporalInstant;
```

temporal-polyfill

https://github.com/fullcalendar/temporal

17.4kB minified + gzipped

```
import { Temporal } from 'temporal-polyfill'

console.log(Temporal.Now.zonedDateTimeISO().toString())
```

Timeline

Temporal is in Stage 3, a final draft before the spec gets approved.

ECMAScript Spec Stages



What doesn’t it have?

- Custom string formatting (You can use Intl.DateTimeFormat for this.)
- Parsing dates from custom strings (e.g. human-readable strings)

How can I use Temporal now?

- Try it in the browser console on https://tc39.es/proposal-temporal/docs/cookbook.html.
- Use a polyfill (not for production).

Further Reading

More Examples

https://tc39.es/proposal-temporal/docs/cookbook.html

https://2ality.com/2021/06/temporal-api.html

Background

https://maggiepint.com/2017/04/09/fixing-javascript-date-getting-started/

Polyfills

https://github.com/js-temporal/temporal-polyfill
https://github.com/fullcalendar/temporal