



VanillaJS et la Web Platform, le couple de l'année?



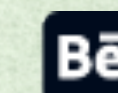
Bertrand Delacrétaz

Principal Scientist, Adobe

Member of the Board of Directors,
Apache Software Foundation

Very Tech Trip, Paris, Février 2023

slides version 2023-02-01b



Artwork by **Alicia Sterling Beach**

Photos are from Adobe Stock unless indicated otherwise



Web Platform?



The Web Platform

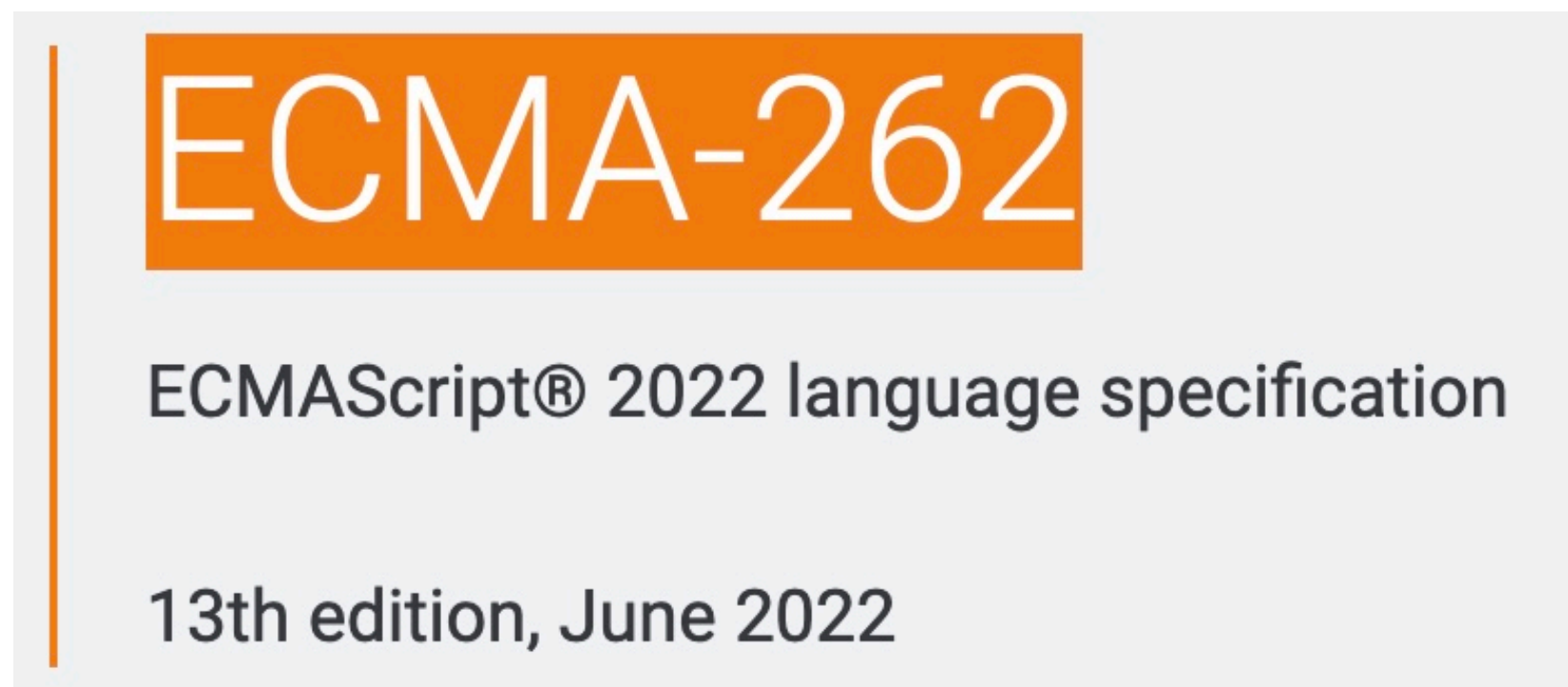
HTML

Living Standard — Last Updated 28 January 2023

<https://html.spec.whatwg.org/>



<https://www.w3.org/Style/CSS/specs.en.html>



<https://www.ecma-international.org/publications-and-standards/standards/ecma-262/>



CURIA VISTA - 1999
www.parlament.ch

Web APIs



When writing code for the Web, there are a large number of Web APIs available. Below is a list of all the APIs and interfaces (object types) that you may be able to use while developing your Web app or site.

<https://developer.mozilla.org/en-US/>

The Web Platform : the Evergreen Web

Constant evolution is fundamental to the web's usefulness



The evergreen Web

W3C TAG Finding 09 February 2017



This Version:

<https://www.w3.org/2001/tag/doc/evergreen-web-20170209>

Latest Version:

<https://www.w3.org/2001/tag/doc/evergreen-web/>

Latest editor's draft:

<https://w3ctag.github.io/evergreen-web/>

Editor:

[Hadley Beeman](#)

Issue Tracking:

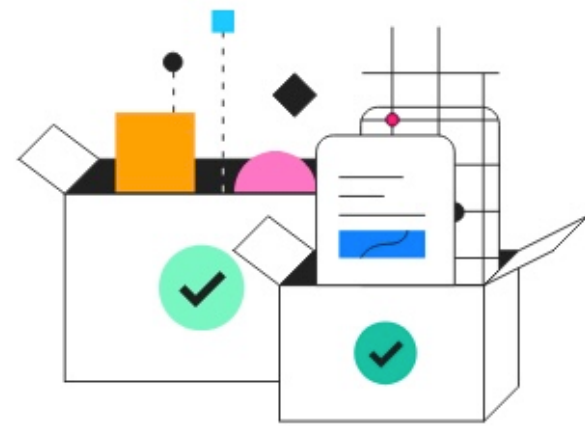
Browsers are a part of the web and therefore they must be continually updated.

<https://www.w3.org/2001/tag/doc/evergreen-web/>

The Web Platform : resources



Mozilla Developer Network (MDN)
<https://developer.mozilla.org/en-US/>



Building a better web, together
Guidance to build modern web experiences that work on any browser.

web.dev
brought to you by the Google Chrome DevRel team
<https://web.dev/>

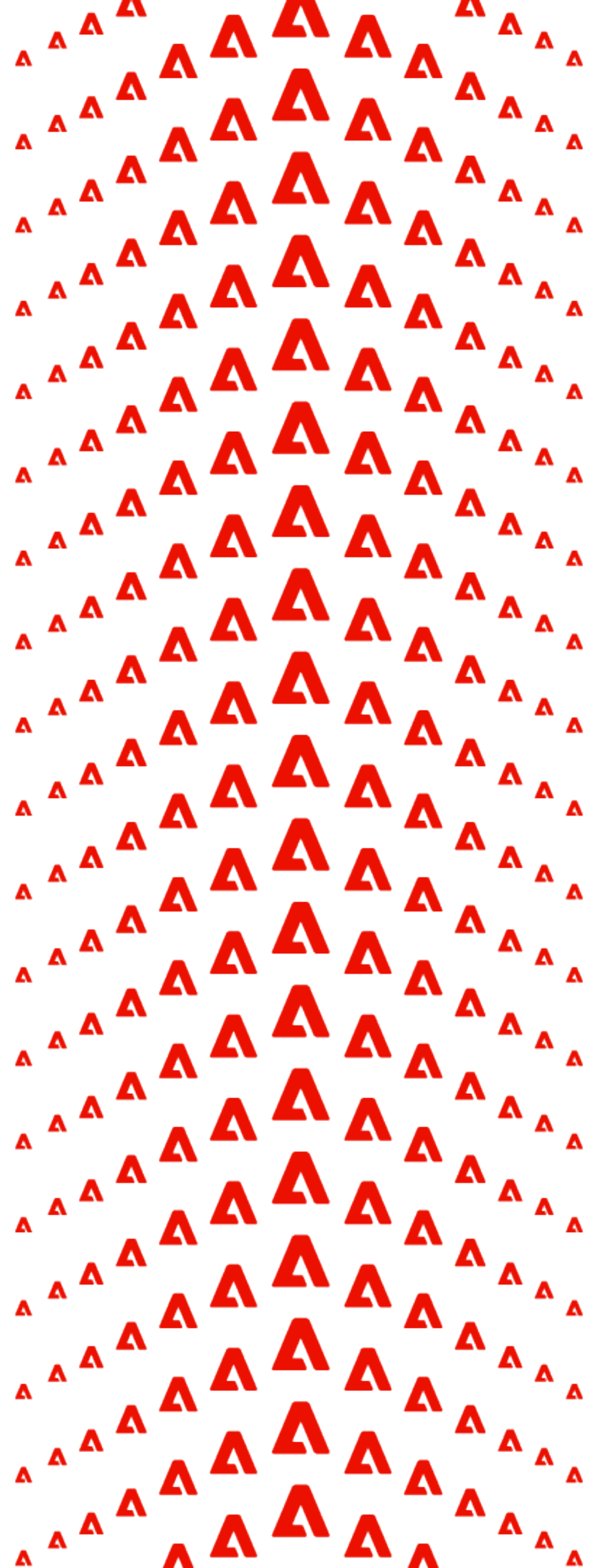


caniuse.com
Browser support tables for modern web technologies



web-platform-tests.org
cross-browser test suite for the Web Platform stack

Vanilla JS ?



Vanilla JS ?

```
// This is the VanillaJS library  
// EOF
```

ECMA-262

ECMAScript® 2022 language specification

13th edition, June 2022

+ Web API

- B**
 - Background Fetch API
 - Background Sync
 - Background Tasks
 - Barcode Detection API
 - Battery API
 - Beacon
- C**
 - CSS Counter Styles
 - CSS Font Loading API
 - CSS Painting API
 - CSS Properties and Values API
 - CSS Typed Object Model API
 - CSSOM
 - Canvas API
 - Channel Messaging API
 - Clipboard API
 - Compression Streams API
- D**
 - DOM
 - Device Orientation Events
- E**
 - Encoding API
 - Encrypted Media Extensions
- F**
 - Fetch API
 - File API
 - File System Access API
 - File and Directory Entries API
- H**
 - HTML DOM
 - HTML Drag and Drop API
 - HTML Sanitizer API
 - History API
- I**
 - Idle Detection API
 - Image Capture API
 - IndexedDB
 - Ink API
- L**
 - Layout Instability API
 - Local Font Access API
- M**
 - Media Capabilities API
 - Media Capture and Streams
 - Media Session API
 - Media Source Extensions
 - MediaStream Recording
 - Bluetooth API
 - Broadcast Channel API
 - Navigation API
 - Network Information API
- P**
 - Console API
 - Contact Picker API
 - Content Index API
 - Cookie Store API
 - Credential Management API
 - Periodic Background Sync
 - Presentation API
 - Permissions API
 - Picture-in-Picture API
 - Pointer Events
 - Pointer Lock API
 - Push API
 - Prioritized Task Scheduling API
 - Proximity Events
- R**
 - Reporting API
 - Resize Observer API
- S**
 - Screen Capture API
 - Screen Orientation API
 - Screen Wake Lock API
 - Sensor API
 - Server Sent Events
 - Service Workers API
 - Storage Access API
 - Streams
- U**
 - UI Events
 - URL API
 - URL Pattern API
 - User-Agent Client Hints API
- V**
 - Vibration API
 - VirtualKeyboard API
 - Visual Viewport
- W**
 - Web Animations
 - Web Share API
 - WebRTC

<https://developer.mozilla.org/en-US/docs/Web/API>

JavaScript Frameworks: why?

components

tools

examples

portable

best practices

well-known

training

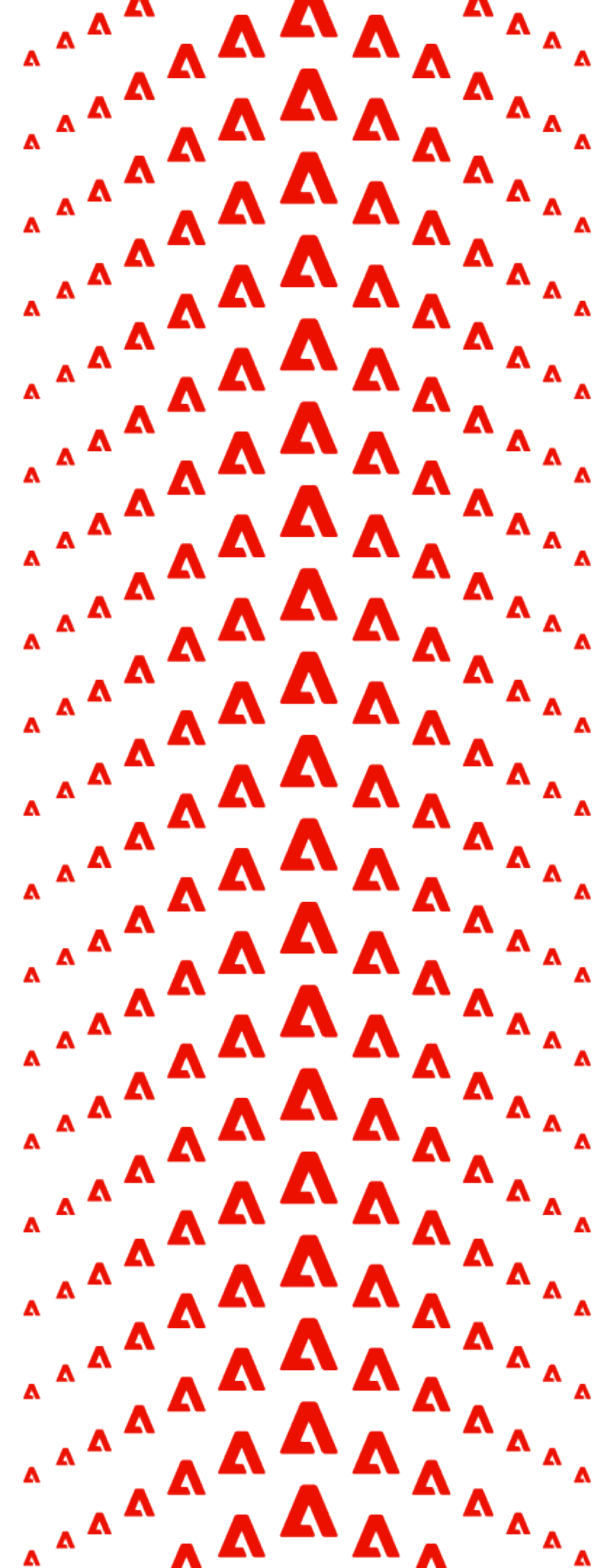
developers

conferences documentation

events

modern

No more Web Frameworks?

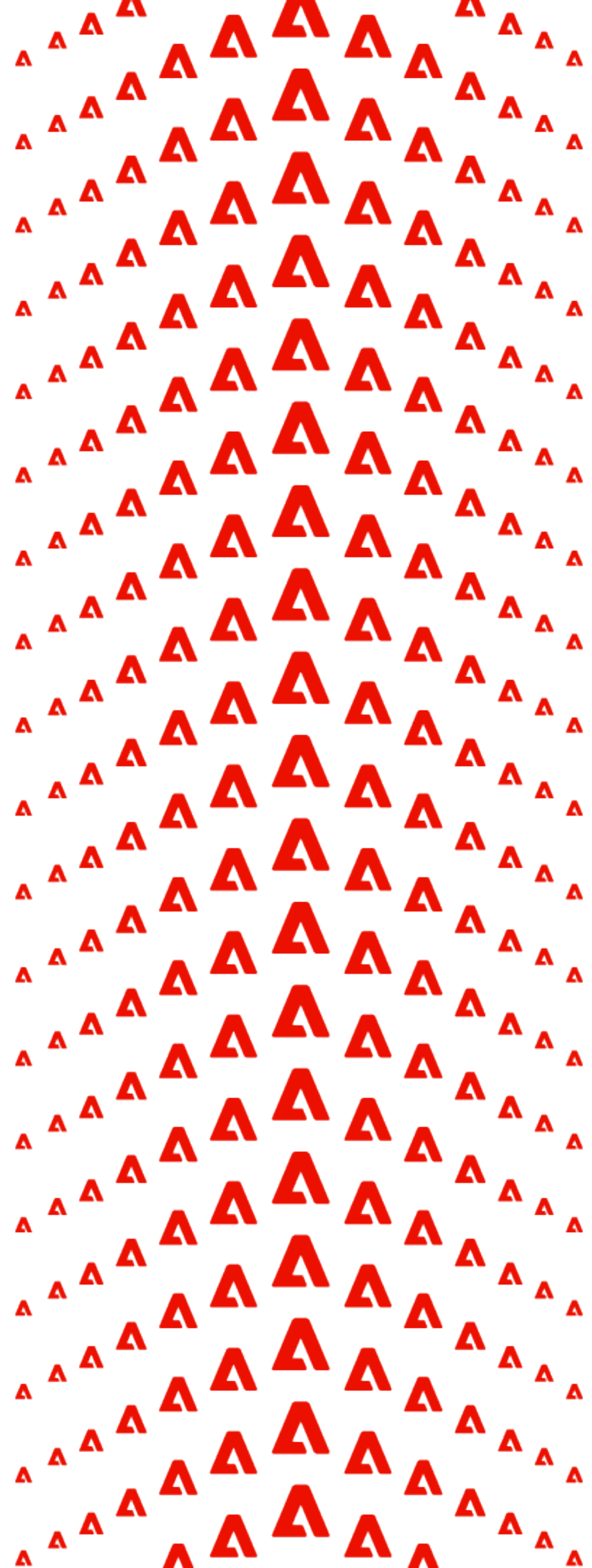


Web Platform, Web Components?



The Web Platform and Web Components provide ~~all 99%~~ 93.42% of this!

Web Components



Web Components ?

MDN says:

Web Components is a suite of different technologies allowing you to create **reusable custom elements** — with their functionality encapsulated away from the rest of your code — and utilize them in your web apps.

https://developer.mozilla.org/en-US/docs/Web/Web_Components

custom elements
shadow dom
HTML templates
encapsulated JavaScript & CSS
progressive enhancement
custom events
polyfills

Web Components : who ?

Spectrum Web Components

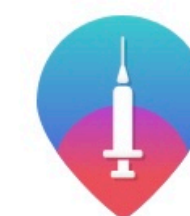
The Spectrum Web Components project is an implementation of [Spectrum, Adobe's design system](#). It's designed to work with any web framework — or even without one.

<https://opensource.adobe.com/spectrum-web-components/>

How we use Web Components at GitHub

At GitHub, we pride ourselves on delivering a first-class developer experience. A considerable part of our work is on our front end, which we strive to keep as lightweight, fast,...

<https://github.blog/2021-05-04-how-we-use-web-components-at-github/>



ViteMaDose
par CovidTracker

À propos

CovidTracker ↗

Votre vaccination contre la Covid **facilement et **rapidement****

Localisation :

Commune, Code postal, Département...

<https://vitemadose.covidtracker.fr/>

Web Components : exemple



ViteMaDose

par CovidTracker

À propos | CovidTracker ↗

Votre vaccination contre la Covid **facilement et **rapidement****

Localisation :

Commune, Code postal, Département...

```
▼ <body> = $0
  ▼ <iframe src="https://www.googletagmanager.com/ns.html?id=GTM-KWWTR1"0" style="display:none;visibility:hidden">
    ▶ #document
    </iframe>
  ▼ <div class="container-xxl">
    ▼ <vmd-app>
      ▶ Shadow Content (Open)
      ▶ <h1 slot="main-title">...</h1>
      ▶ <noscript id="noscript">...</noscript>
      ▶ <div slot="about" style="margin-top: 40px; margin-left: 10px">.
      ▶ <div slot="about-lieux">...</div>
      ▶ <div slot="chronodose">...</div>
    </vmd-app>
```

Votre vaccination contre la Covid **facilement et **rapidement****



Vous utilisez un navigateur internet obsolète.

Veillez mettre à jour votre navigateur pour améliorer votre expérience et votre sécurité en cliquant ici.

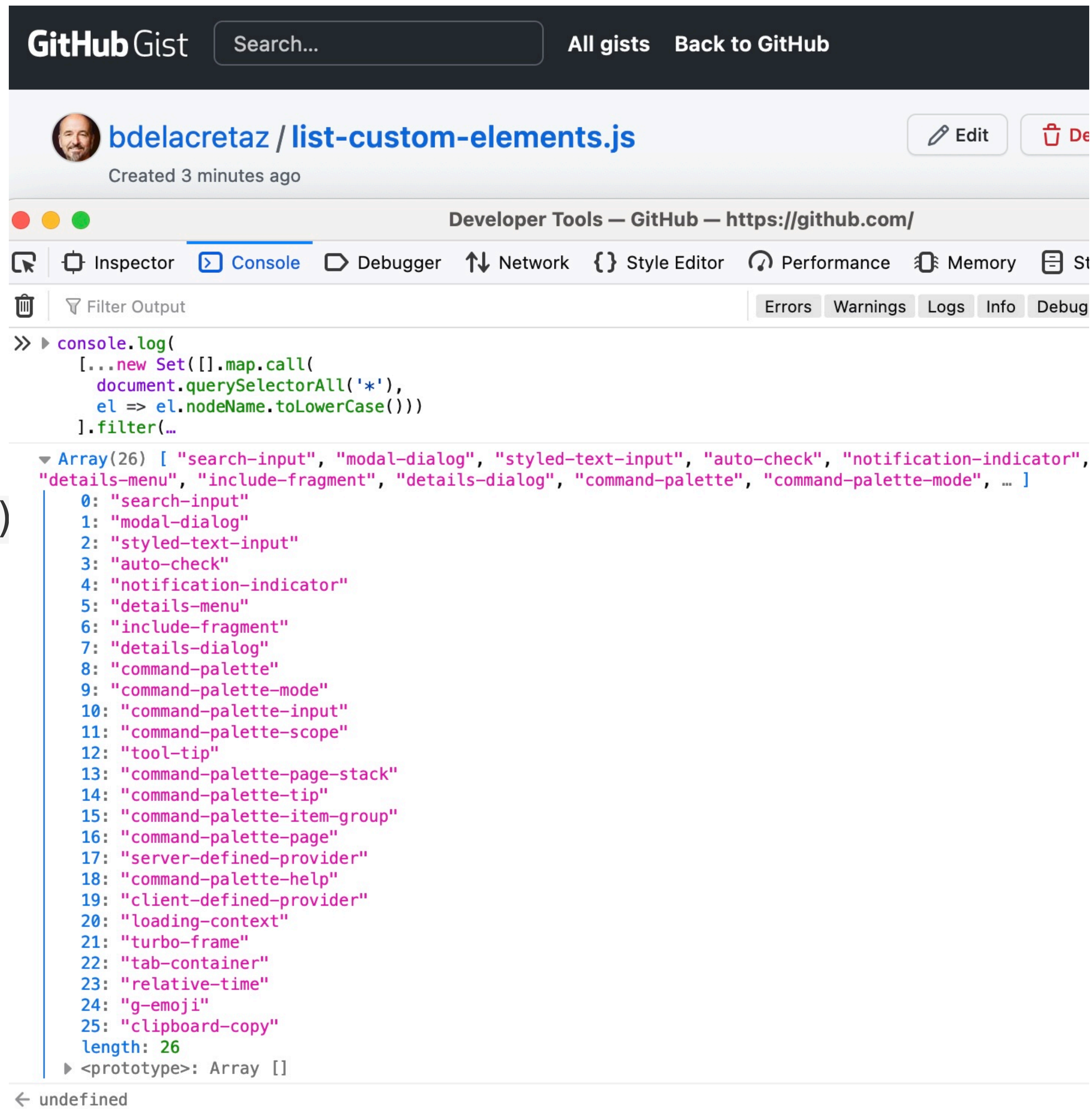
no JavaScript:

Qu'est-ce que Vite Ma Dose ?

Vite Ma Dose est un outil de CovidTracker permettant de détecter les rendez-vous de vaccination contre la Covid-19. Vite Ma Dose propose des rendez-vous disponibles dans votre département afin de vous faire vacciner (sous réserve d'éligibilité). Vite Ma Dose offre une solution rapide, simple et fiable pour trouver un rendez-vous de vaccination contre la Covid-19.

Web Components : list custom element names

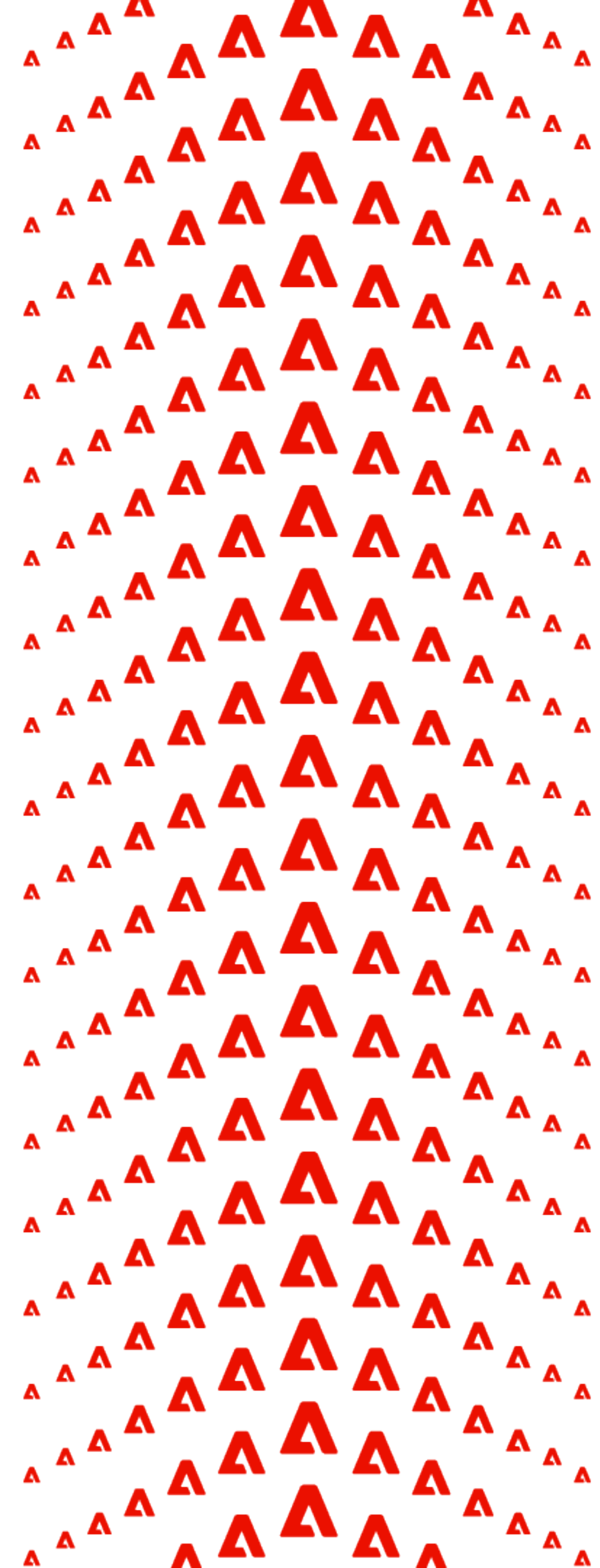
```
console.log(  
  [...new Set([].map.call(  
    document.querySelectorAll('*'),  
    el => el.nodeName.toLowerCase())  
  ].filter(  
    customElements  
      .get.bind(customElements)  
  )  
)
```



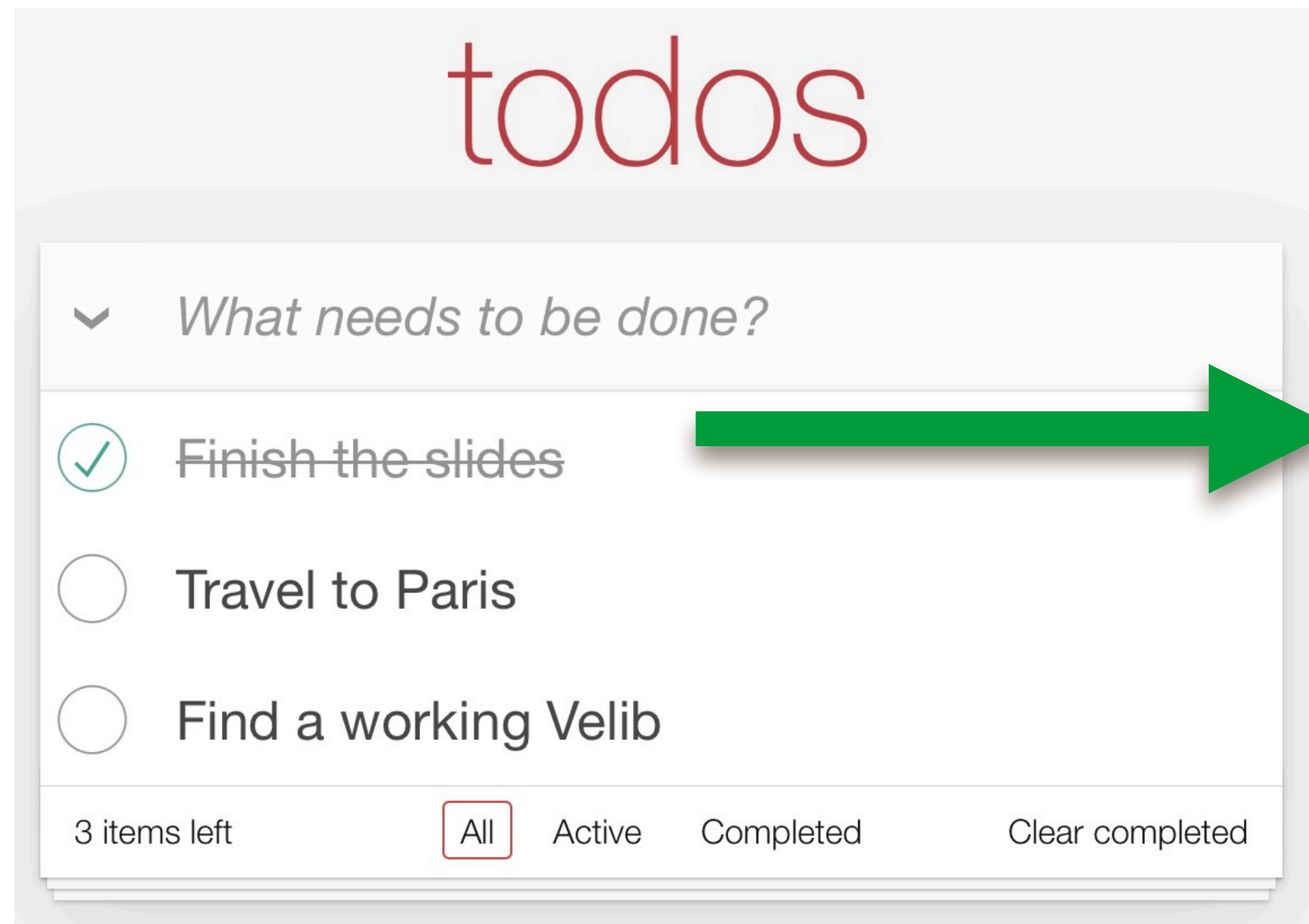
The screenshot shows a GitHub Gist page for a user named bdelacretaz, titled "list-custom-elements.js". The gist was created 3 minutes ago. Below the gist title, a browser's developer tools console is open, displaying the output of the JavaScript code. The console shows a log message with an array of 26 custom element names, including "search-input", "modal-dialog", "styled-text-input", "auto-check", "notification-indicator", "details-menu", "include-fragment", "details-dialog", "command-palette", "command-palette-mode", "command-palette-input", "command-palette-scope", "tool-tip", "command-palette-page-stack", "command-palette-tip", "command-palette-item-group", "command-palette-page", "server-defined-provider", "command-palette-help", "client-defined-provider", "loading-context", "turbo-frame", "tab-container", "relative-time", "g-emoji", and "clipboard-copy". The array has a length of 26. The console also shows the prototype of the array as "Array []".

TodoMVC

Using Web Components



TodoMVC using Web Components



```
<todo-list>
  <ul class="todo-list">
    <todo-item id="id_1675256271629_1675256271630" completed="true">
      <Shadow Content (Closed)>
        <style>...</style>
        <li class="completed">
          <div class="view">
            <input class="toggle" type="checkbox" checked>
            <label>Finish the slides</label>
            <button class="destroy"></button>
          </div>
          <input data-todo="edit" class="edit">
        </li>
      </todo-item>
    <todo-item id="id_1675256271630_1675256271630">...</todo-item>
    <todo-item id="id_1675256271631_1675256271630">...</todo-item>
  </ul>
</todo-list>
```

<https://github.com/adobe/web-platform-zoo>

<https://opensource.adobe.com/web-platform-zoo/examples/web-components/todomvc/>

TodoMVC: todo-list component

```
class TodoListWebComponent extends HTMLElement {
  #listRoot;

  connectedCallback() {
    // called when the element is inserted in the DOM
    this.#listRoot = document.createElement("ul");
    this.#listRoot.setAttribute("class", "todo-list");
    this.append(this.#listRoot);
    window.todomvc.store.addEventListener('save', this._render.bind(this));
    window.addEventListener('todomvc-filter', this._render.bind(this));
    this._render();
  }

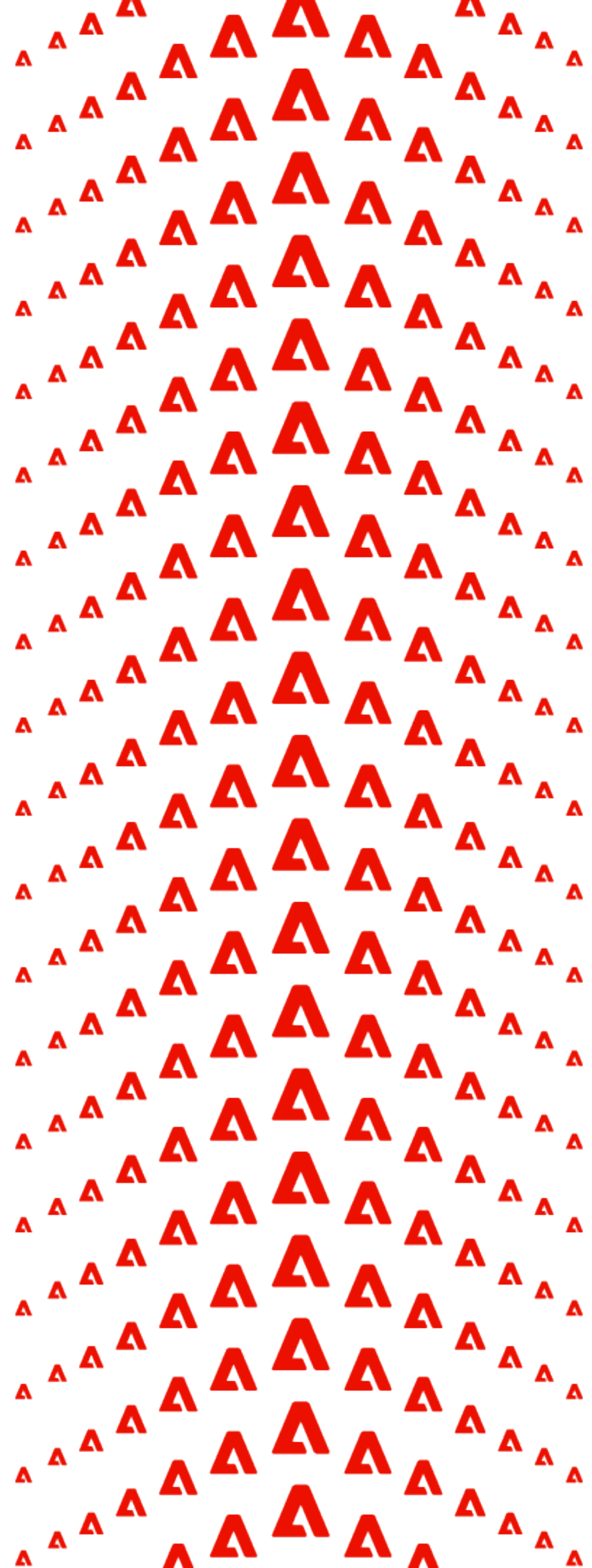
  _render() {
    const currentFilter = this._getFilter(document.location.hash);
    this.#listRoot.innerHTML = '';
    for(let todo of window.todomvc.store.all(currentFilter)) {
      // create custom elements for the list items
      const e = document.createElement('todo-item');
      e.setAttribute('id', todo.id);
      if(todo.completed) {
        e.setAttribute('completed', 'true');
      }
      e.textContent = todo.title;
      this.#listRoot.appendChild(e);
    };
  }
}
```

Usage:

```
<section data-todo="hide-if-none" class="main">
  <input id="toggle-all" class="toggle-all" type="checkbox">
  <label for="toggle-all">Mark all as complete</label>
  <todo-list/>
</section>
```

```
customElements.define('todo-list', TodoListWebComponent);
```

Testing



End-to-end testing: Playwright ?



Playwright

<https://playwright.dev/>

Cross-browser, cross-platform, cross-language

```
const rockButton = page.locator('li:has-text("Rock Climbing")');
await rockButton.click();
expect(await countCards()).toEqual(2);
await assertTitlesPresent([
  "Climbing New Zealand",
  "Overnight Colorado Rock Climbing"
])
```

Testing the TodoMVC Web Components

```
test("Add & count items", async ({ page }) => {
  const w = new Wrapper(page);
  await w.addItem("First One");
  await w.assertCountDisplay(initialItemCount + 1);
  await w.addItem("Second Two");
  await w.assertCountDisplay(initialItemCount + 2);
  await expect(w.filtersAndStatus).toBeVisible();
});
```

```
test("Edit the second item", async ({ page }) => {
  const w = new Wrapper(page);

  const oldText = initialItems[1];
  const addedText = 'rulesrules'
  const newText = oldText + addedText;

  await w.assertItemValue(1, oldText);
  await w.secondItem.dblclick();
  await w.secondItem.type(addedText);
  await w.secondItem.press('Enter');
  await w.assertCountDisplay(initialItemCount);
  await w.assertItemValue(1, newText);
});
```

Playwright test report:

✓ web-components/todomvc/tests/todomvc.test.js	8.2s
✓ TodoMVC using Web Components > Add & count items web-components/todomvc/tests/todomvc.test.js:75	chromium 267ms
✓ TodoMVC using Web Components > Add & count items web-components/todomvc/tests/todomvc.test.js:75	firefox 345ms
✓ TodoMVC using Web Components > Add & count items web-components/todomvc/tests/todomvc.test.js:75	webkit 383ms
✓ TodoMVC using Web Components > Add & count it... web-components/todomvc/tests/todomvc.test.js:75	Mobile Chrome 236ms
✓ TodoMVC using Web Components > Add & count ite... web-components/todomvc/tests/todomvc.test.js:75	Mobile Safari 360ms
✓ TodoMVC using Web Components > Add & count it... web-components/todomvc/tests/todomvc.test.js:75	Microsoft Edge 387ms
✓ TodoMVC using Web Components > Add & count i... web-components/todomvc/tests/todomvc.test.js:75	Google Chrome 234ms

Coda



Web Components

modern and forward-looking

modern css standards

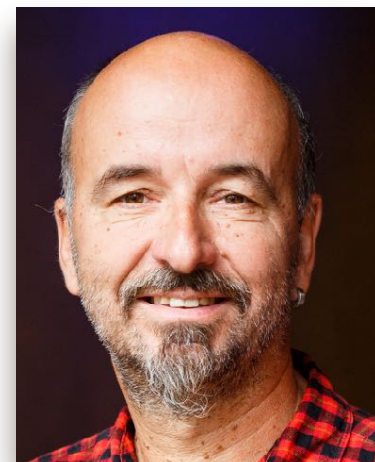
the Web Platform is back!

modern html

Evergreen Web

modern JavaScript

long-lived learning



@bdelacretaz
<https://fosstodon.org/@bdelacretaz>
<https://grep.codeconsult.ch/>



 Published in Adobe Tech Blog

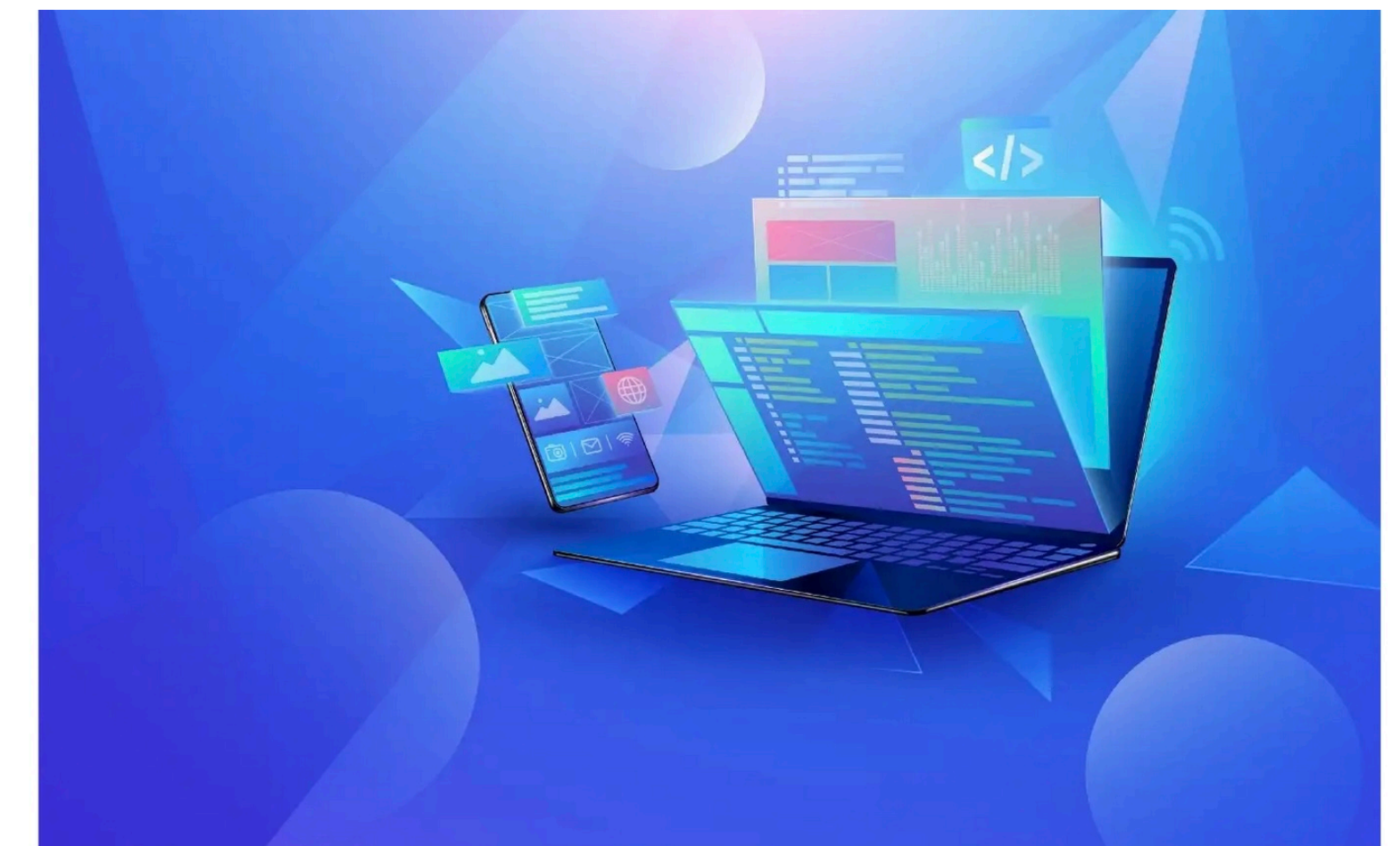


Bertrand Delacretaz

Dec 14, 2022 · 5 min read · Listen



The Web Platform Is Back



<https://blog.developer.adobe.com/the-web-platform-is-back-fa5752fabdfc>