

WEB COMPONENTS CAN'T SAVE US.

But you can!

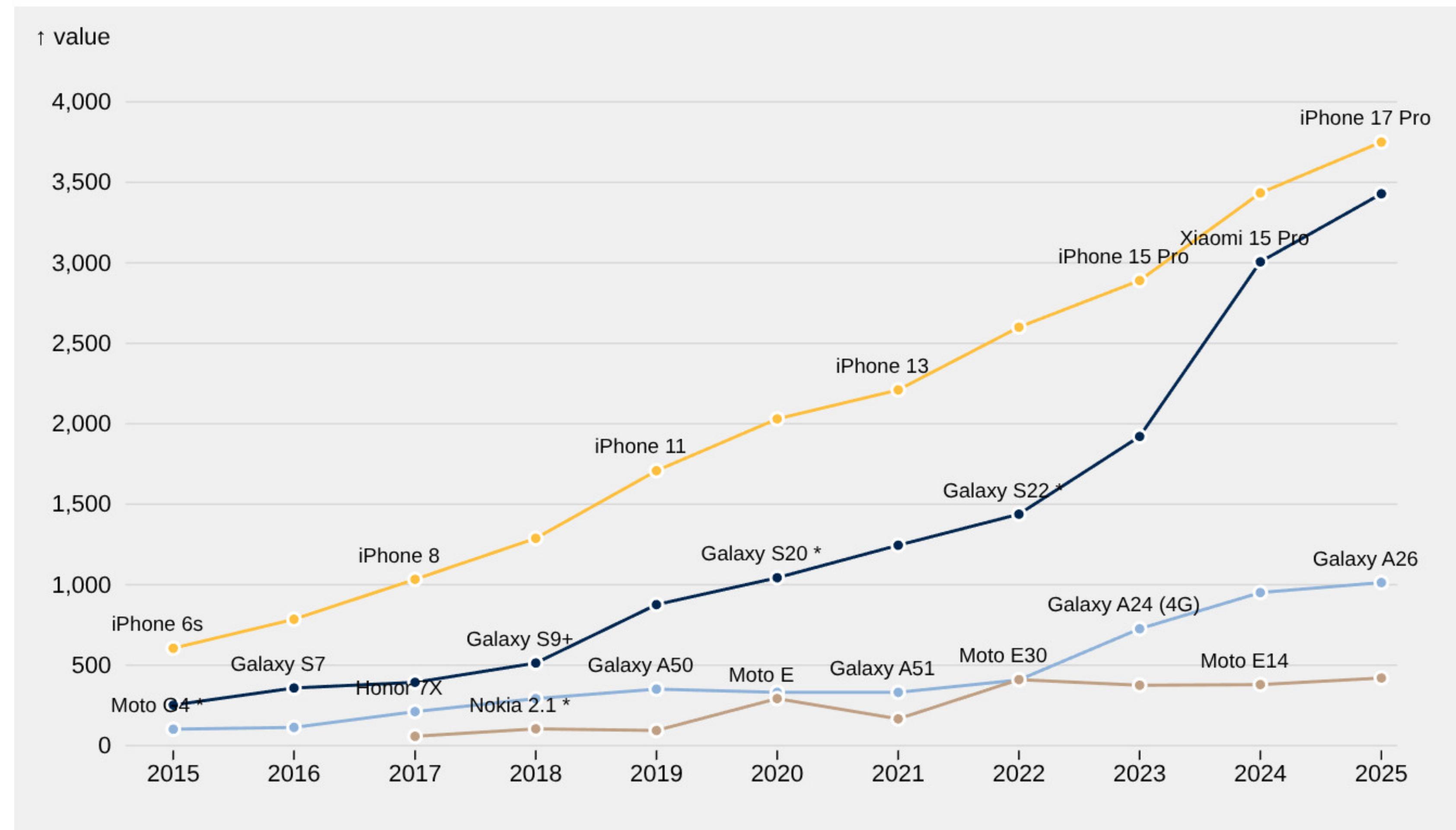
Scott Jehl

Squarespace Web Performance

the future is here...

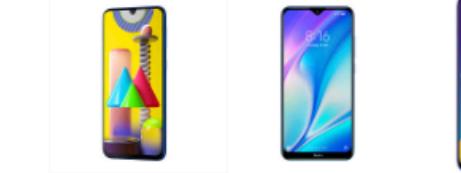
Geekbench 5 Single-Core Scores

■ Android Single-Core ■ Budget Android Single-Core ■ Low-End Single-Core ■ iOS Single-Core



Best Sellers in Cell Phones

#1		<p>Apple iPhone 12, 64GB, Black - Fully Unlocked (Renewed) ★★★★★ 32,455 \$204.95</p>
#2		<p>Samsung Galaxy A16 5G A Series, Unlocked Android Smartphone, Large AMOLED Display, Durable Design, Super Fast Charging,... ★★★★★ 2,521 \$149.99</p>
#3		<p>Apple iPhone 14, 128GB, Midnight - Unlocked (Renewed) ★★★★★ 5,102 \$310.64</p>
#4		<p>Motorola Moto G Play 2024 Unlocked Made for US 4/64GB 50MP Camera Sapphire Blue ★★★★★ 3,720 \$99.99</p>
#5		<p>Apple iPhone 13, 128GB, Midnight - Unlocked (Renewed) ★★★★★ 19,272 \$292.08</p>
#6		<p>SAMSUNG Galaxy S25 Edge Phone, 512 GB AI Smartphone, Unlocked Android, Night Video, Fast Processor, ProScaler Display, All-Day... ★★★★★ 3,464 \$819.99</p>
#7		<p>Apple iPhone 11, 64GB, Black - Unlocked (Renewed) ★★★★★ 57,006 \$172.99</p>
#8		<p>Motorola Moto G Stylus - 2025 Unlocked Made for US 8/256GB 50MP Camera Pantone Gibraltar Sea ★★★★★ 2,089 \$284.99</p>

#1	#2	#3	#4	#5	#6
					
iQOO Z10 Lite 5G (Titanium Blue, 6GB RAM, 128GB Storage) 6000 mAh Battery Dimensity 6300 5G Processor with 433K+... ★★★★★ 1,341 ₹10,998.00	iQOO Z10 Lite 5G (Cyber Green, 6GB RAM, 128GB Storage) 6000 mAh Battery Dimensity 6300 5G Processor with 433K+... ★★★★★ 1,341 ₹10,998.00	Samsung Galaxy M05 (Mint Green, 4GB RAM, 64 GB Storage) 50MP Dual Camera Bigger 6.7" HD+ Display 5000mAh Battery 25W... ★★★★★ 5,917 ₹6,499.00	iQOO Z10x 5G (Ultramarine, 6GB RAM, 128GB Storage) 6500 mAh Large Capacity Battery Dimensity 7300 Processor Military-Grade... ★★★★★ 3,290 ₹13,498.00	iQOO Z10 Lite 5G (Titanium Blue, 4GB RAM, 128GB Storage) 6000 mAh Battery Dimensity 6300 5G Processor with 433K+... ★★★★★ 1,341 ₹9,998.00	iQOO Z10 Lite 5G (Cyber Green, 4GB RAM, 128GB Storage) 6000 mAh Battery Dimensity 6300 5G Processor with 433K+... ★★★★★ 1,341 ₹9,998.00
					
iQOO Neo 10 (Titanium Chrome, 8GB RAM, 256GB Storage) Snapdragon 8s Gen 4 Processor & SuperComputing Chip Q1 7000... ★★★★★ 1,681 ₹33,998.00	OnePlus Nord CE4 Lite 5G (Super Silver, 8GB RAM, 128GB Storage) Lifetime Display Warranty 5500 mAh Battery, 80W... ★★★★★ 8,664 ₹16,999.00	iQOO Z10x 5G (Titanium, 6GB RAM, 128GB Storage) 6500 mAh Large Capacity Battery Dimensity 7300 Processor Military-Grade... ★★★★★ 3,290 ₹13,498.00	Redmi A4 5G (Sparkle Purple, 4GB RAM, 64GB Storage) Global Debut SD 4s Gen 2 Segment Largest 6.88in 120Hz 50MP Dual Camera ... ★★★★★ 5,821 ₹7,998.00	Redmi A4 5G (Starry Black, 4GB RAM, 64GB Storage) Global Debut SD 4s Gen 2 Segment Largest 6.88in 120Hz 50MP Dual Camera ... ★★★★★ 5,821 ₹7,998.00	iQOO Neo 10 (Titanium Chrome, 12GB RAM, 256GB Storage) Snapdragon 8s Gen 4 Processor & SuperComputing Chip Q1 7000... ★★★★★ 1,681 ₹35,998.00
					
Hot New Releases in Smartphones		Most Wished For in Smartphones		Most Gifted in Smartphones	
#13	#14	#15	#16	#17	#18
					

History of Web Performance

— A TIMELINE —

**Everything is slow
due to constraints**



1989



2013

**Everything is slow due to
developers' reduced constraints**



Present

A photograph of a man in a blue shirt pouring olive oil from a large glass jug into a bowl of salad. The scene is set in a rustic kitchen with wooden barrels in the background.

JAVASCRIPT

DEVELOPERS

WEBSITE

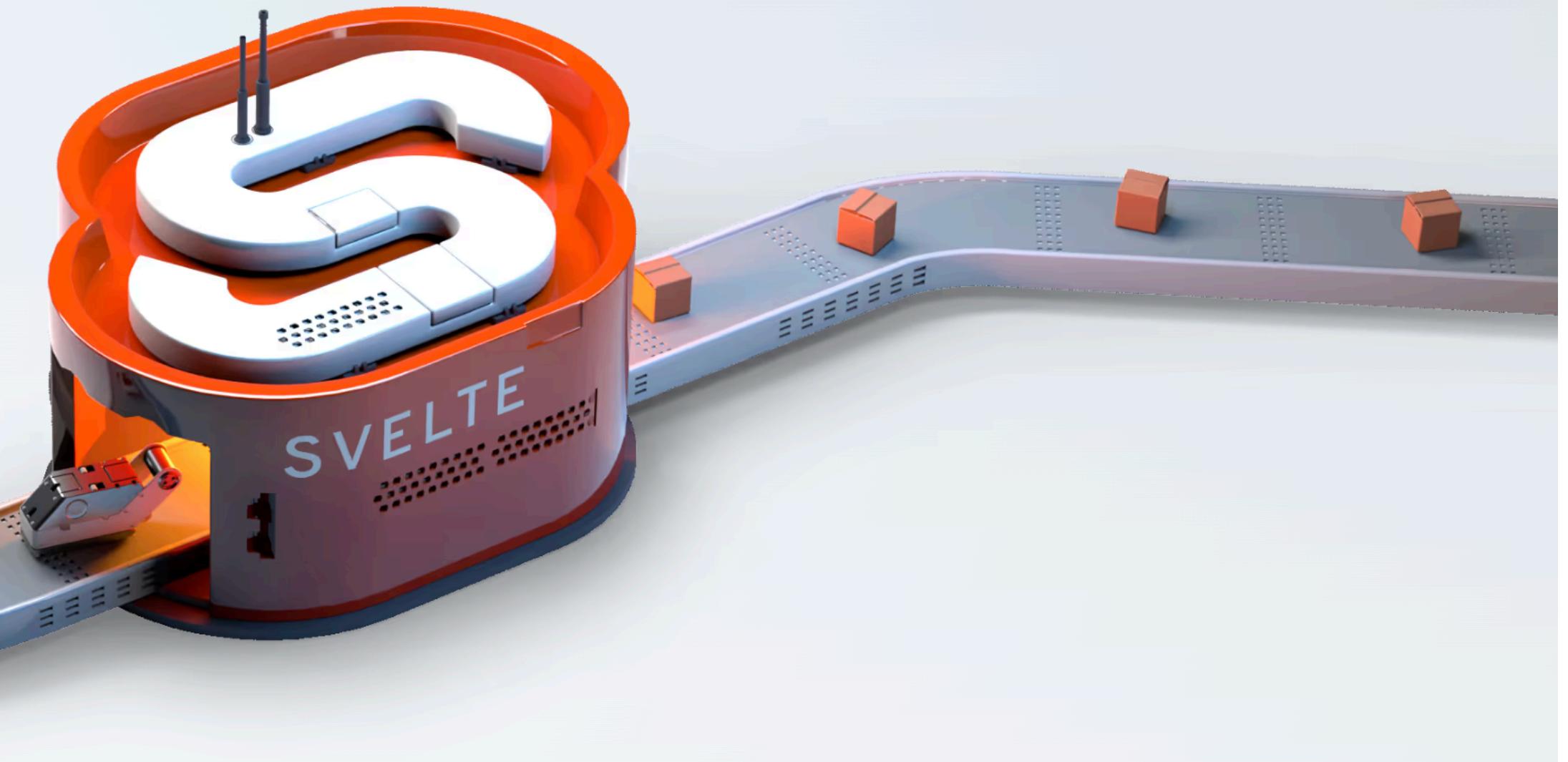




we shifted the burden
from the network to the device

web development for the rest of us

[GET STARTED →](#)

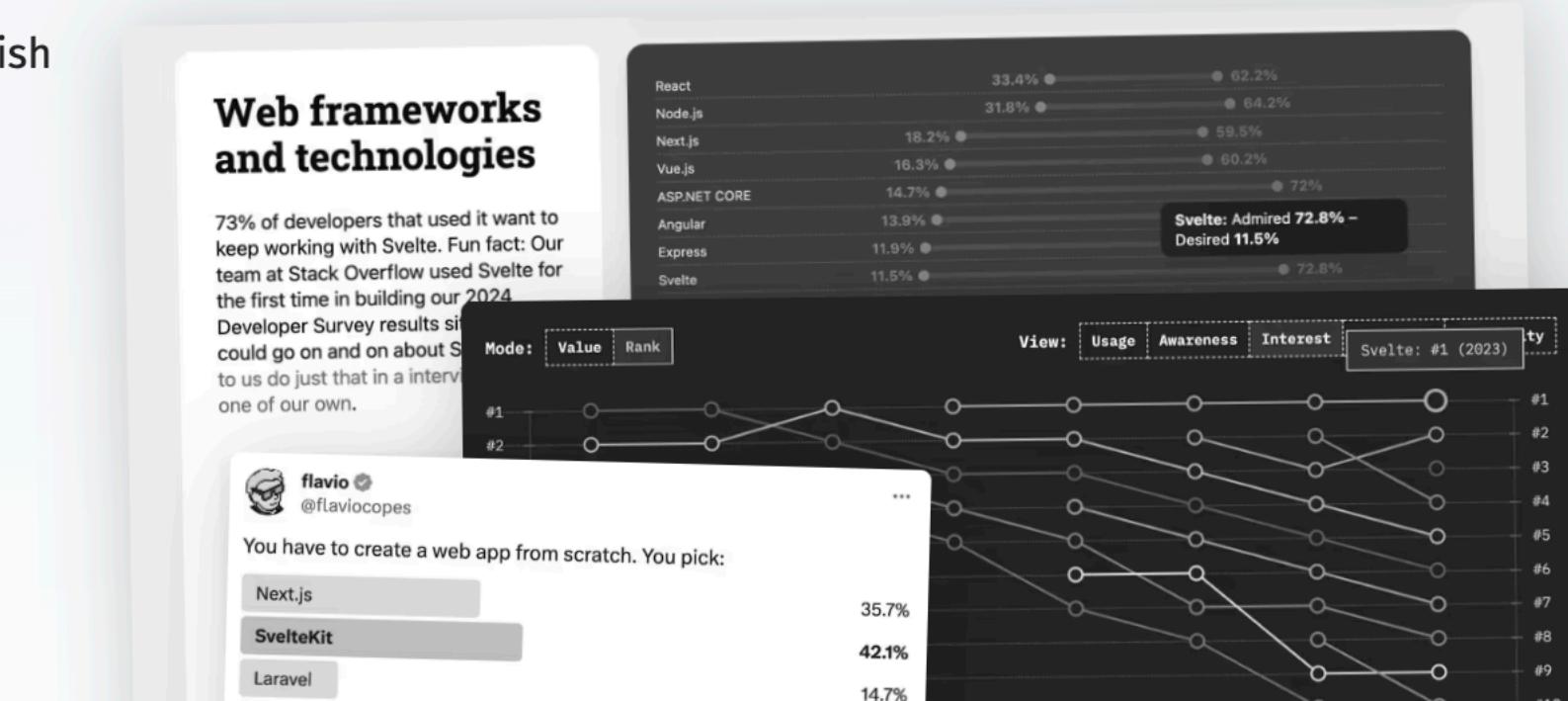


/'svɛlt/

ADJECTIVE attractively thin, graceful and stylish

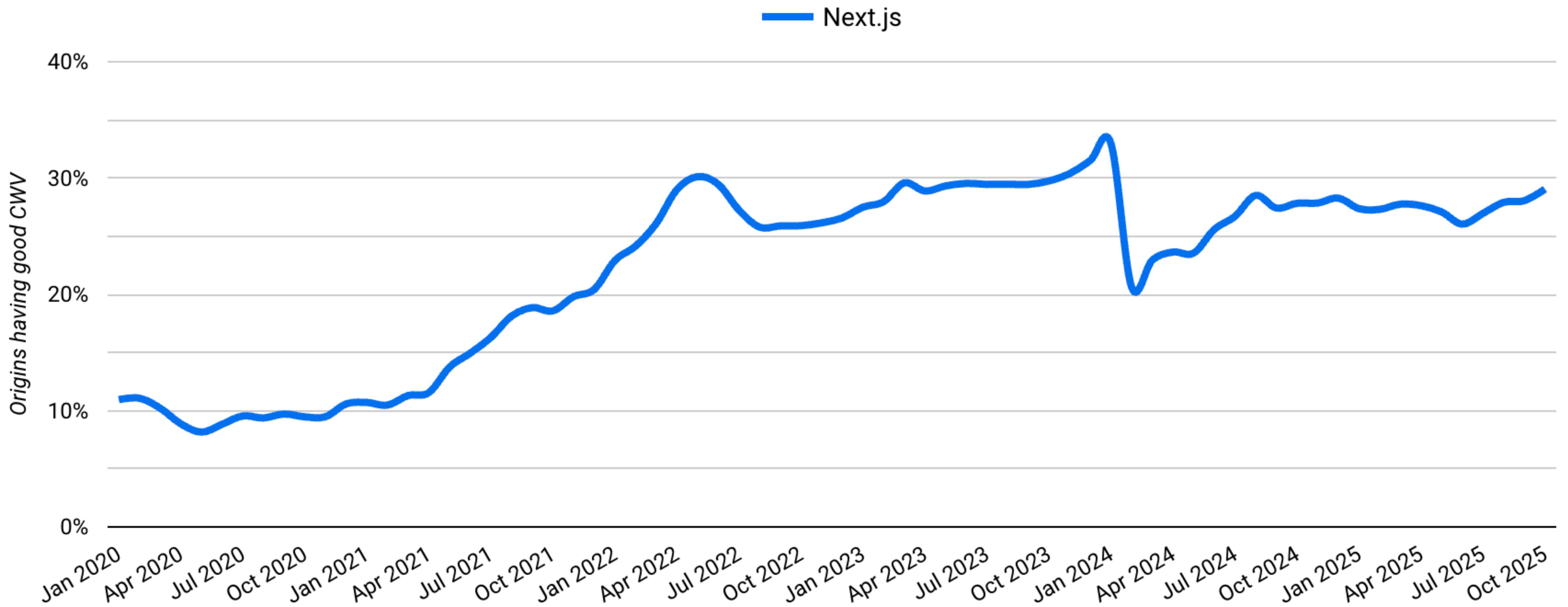
Svelte is a UI framework that uses a compiler to let you write breathtakingly concise components that do minimal work in the browser, using languages you already know — HTML, CSS and JavaScript. It's a love letter to web development.

But don't take our word for it. Developers consistently rank



Rank: ALL

Geo: ALL





A close-up photograph of the back of a dark-colored Motorola Moto G Power 2022 smartphone. The phone features a triple-camera system on the left side, consisting of three circular lenses and a small flash. On the right side, there is a white rectangular sticker with the text "MOTOG POWER" and "2022 LIGHTHOUS" printed on it. The Motorola logo is visible at the bottom center. The phone is resting on a light-colored wooden surface.

MOTOG POWER
2022 LIGHTHOUS



Charles

WEB DEBUGGING PROXY APPLICATION
for Windows, Mac OS and Linux

[HOME](#)[OVERVIEW](#)[DOCUMENTATION](#) [DOWNLOAD](#)[BUY](#)[SUPPORT](#)

Charles is an HTTP proxy / HTTP monitor / Reverse Proxy that enables a developer to view all of the HTTP and SSL / HTTPS traffic between their machine and the Internet. This includes requests, responses and the HTTP headers (which contain the cookies and caching information).

[READ MORE](#)

The screenshot shows the Charles 4.0 interface with the title bar "Charles 4.0 - Session 1". The left sidebar lists network sessions: "https://www.charlesproxy.com" (selected), "static" (expanded, showing "img" folder with files like "hdr_reviews.ea40c8b6.png", "hdr_news.21990fa4.png", "hdr_features.7ab7d072.png", and "icon_globe.c55491a0.png"), "assets", and "<default>". Below the sidebar is a table with columns "Overview", "Contents", "Summary", "Chart", and "Notes", with "Contents" being the active tab. The main pane displays a "GET / HTTP/1.1" request for "Host: www.charlesproxy.com". It shows detailed headers: "Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8", "Cookie: JSESSIONID=8510ABC0AE3DBE91846AB9013F94784E; content_tree_open...", "User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_5) AppleWebKit/601.6.17 (KHTML, like Gecko)", "Accept-Language: en-us", "Accept-Encoding: gzip, deflate", and "Connection: keep-alive". Below the headers is the raw HTML response, which includes the DOCTYPE declaration, XML declarations, and the start of the HTML document.

<https://www.charlesproxy.com/>

DOWNLOAD



[Download a free trial](#)
Version 5.0.2



Check out the latest remote job listings from Authentic Jobs.

ads via Carbon

core web vitals

(Loading)

LCP

Largest Contentful Paint



(Interactivity)

INP

Interaction to Next Paint



(Visual Stability)

CLS

Cumulative Layout Shift





Google Lighthouse



PageSpeed Insights



DebugBear



SpeedCurve

SPEED TEST

Overview

Lighthouse Audits

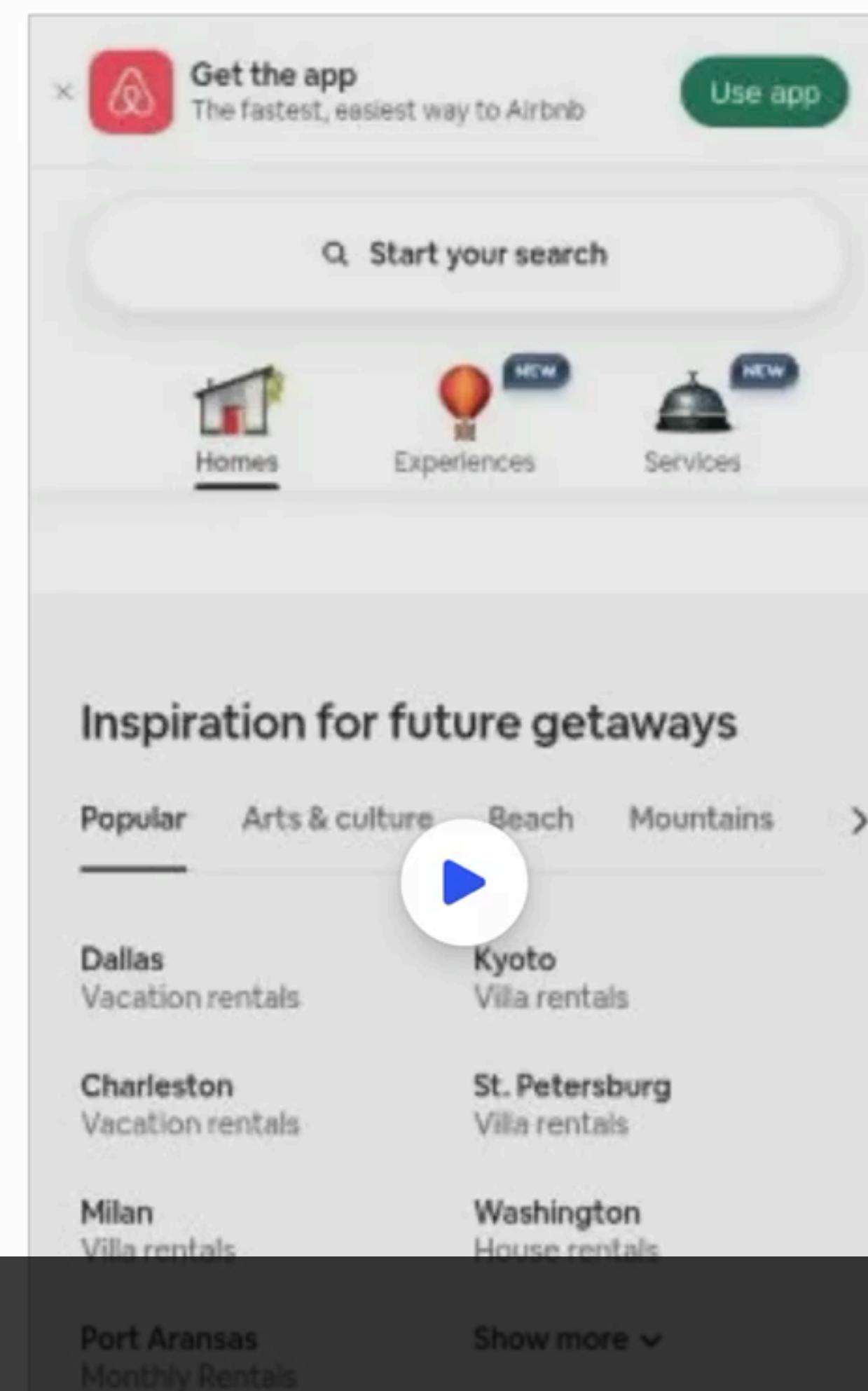
Third Parties

Monitoring

CRUX

 / Page speed test / www.airbnb.com/

🇺🇸 North Virginia · PageSpeed Mobile · Simulated throttling · Chrome 134 & Lighthouse 12.5.1 · Today at 5:57PM

Test summary ([View page ↗](#))

WEB VITALS ASSESSMENT: FAILED

11.65 secLargest Contentful Paint [?](#)**0**Cumulative Layout Shift [?](#)**487** msTotal Blocking Time [?](#)

OTHER NOTABLE METRICS

692 msTime to First Byte [?](#)**258** msResponse Time [?](#)**2.16** secFirst Contentful Paint [?](#)**6.91** secSpeed Index [?](#)<https://calibreapp.com/>

Home >  Homepage UK (Desktop) > 12 Jun 2025, 07:40  



Compare

Last month 

Lab Tests

Page Details

Overview

Web Vitals

Requests

Metrics

Lighthouse

Console

RUM

Experiments

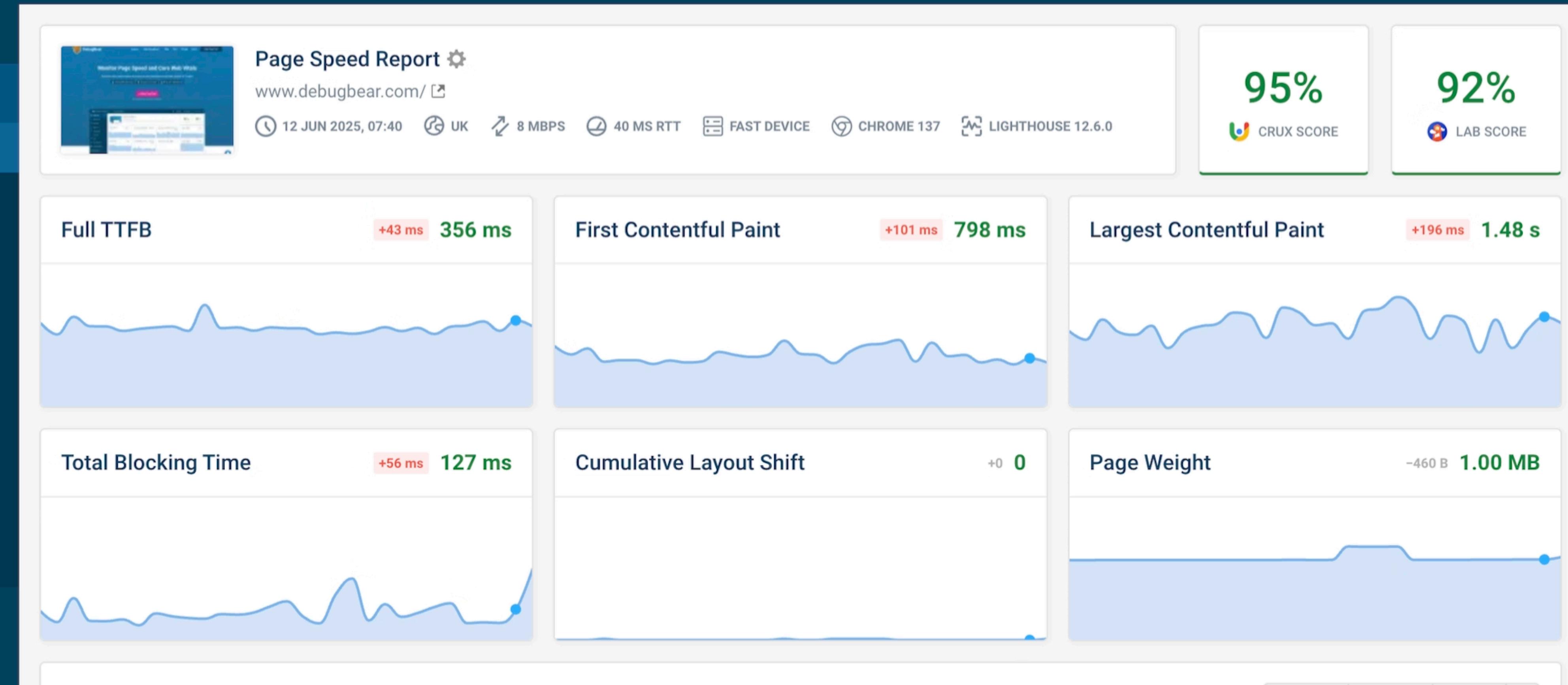
Page Settings

CrUX Trends

Real User Monitoring

Test Page Now

Show Annotations

<https://www.debugbear.com/>

real user monitoring

Speed Test

Run tests with synthetic data to identify opportunities to improve performance.

[Speed Test documentation](#)

URL: taskstar.org/blog, Region: Jurong West, Singapore, Frequency: Daily, Run daily test

You can run 49 more tests this month and schedule 14 more recurring tests.

Page	Region	Frequency	Last test	Score
www.taskstar.org/blog	Iowa, USA	Daily	10 minutes ago	100
taskstar.org/	Sydney, Australia	-	21 days ago	97
taskstar.org/	South Carolina, USA	-	1 day ago	92

[+ Back to Observatory](#)

taskstar.org/

Synthetic Real user measurements

Region: São Paulo, Brazil Test date: 06/19/2023 4:41 PM [View history report](#)

[Run daily test](#) [Run test once](#)

Speed test result

Recommendations

Lighthouse identified the below opportunities. Check the following recommendations to improve your web performance.

Description	Recommendations
Eliminate render-blocking resources	Rocket Loader™
Lighthouse Audit	Resources are blocking the first paint of your page. Consider delivering critical JS/CSS inline and deferring all non-critical JS/styles. Learn how to eliminate render-blocking resources.
Recommendations	Defer your JS until your page has fully loaded with Rocket Loader. Rocket Loader prioritizes your website's content (text, images, fonts, and more) by deferring the loading of all of your JavaScript until after rendering. Learn more about Rocket Loader.
Properly size images	Image Resizing
Minify CSS	Auto Minify
Minify JavaScript	Auto Minify
Efficiently encode images	Polish Enabled
Serve images in next-gen formats	Polish Enabled
Lighthouse Audit	Image formats like WebP and AVIF often provide better compression than PNG or JPEG, which means faster downloads and less data consumption. Learn more about modern image formats.
Recommendations	Compress and encode your images more effectively. Polish strips metadata from images and reduces image size through lossy or lossless compression to accelerate the speed of image downloads. Learn more about Polish.
Initial server response time was short	Argo Enabled
Minimize third-party usage	Zaraz

Recommendations

Lighthouse identified the below opportunities. Check the following recommendations to improve your web performance.

Description	Recommendations
Eliminate render-blocking resources	Rocket Loader™
Lighthouse Audit	Resources are blocking the first paint of your page. Consider delivering critical JS/CSS inline and deferring all non-critical JS/styles. Learn how to eliminate render-blocking resources.
Recommendations	Defer your JS until your page has fully loaded with Rocket Loader. Rocket Loader prioritizes your website's content (text, images, fonts, and more) by deferring the loading of all of your JavaScript until after rendering. Learn more about Rocket Loader.

[← Back to Speed Test](#)

www.taskstar.org/blog

Desktop Mobile

Synthetic Real user measurements

Real user measurements

Real user measurements show the end users experience on your website.

Previous 7 days

Time To First Byte	0.52 s
First Contentful Paint	0.95 s
Largest Contentful Paint	1.39 s
Cumulative Layout Shift	0.026
Interaction to Next Paint	176 ms
First Input Delay	39 ms

Interaction to Next Paint by country

www.taskstar.org/blog

Region: Iowa, USA

Trend

Trend shows the result of score and metrics over time for you to easily track the website performance.

Last 7 days

Performance score TTFB FCP SI LCP TTI TBT CLS

History report

Test date	Frequency	Status	TTFB	FCP	LCP	TTI	TBT	SI	CLS
7 Jun, 1:48 PM	Daily	100	0.06 s	0.35 s	0.66 s	0.40 s	0 ms	0.41 s	0
6 Jun, 1:48 PM	Daily	98	0.04 s	0.71 s	0.97 s	0.71 s	0 ms	0.97 s	0

<https://blog.cloudflare.com/cloudflare-observatory-generally-available/>

CrUX



Chrome UX Report

Technology

Squarespace

Client

Mobile

Squarespace provides Software-as-a-Service (SaaS) for website building and hosting, and allows users to use pre-built website templates.

CMS

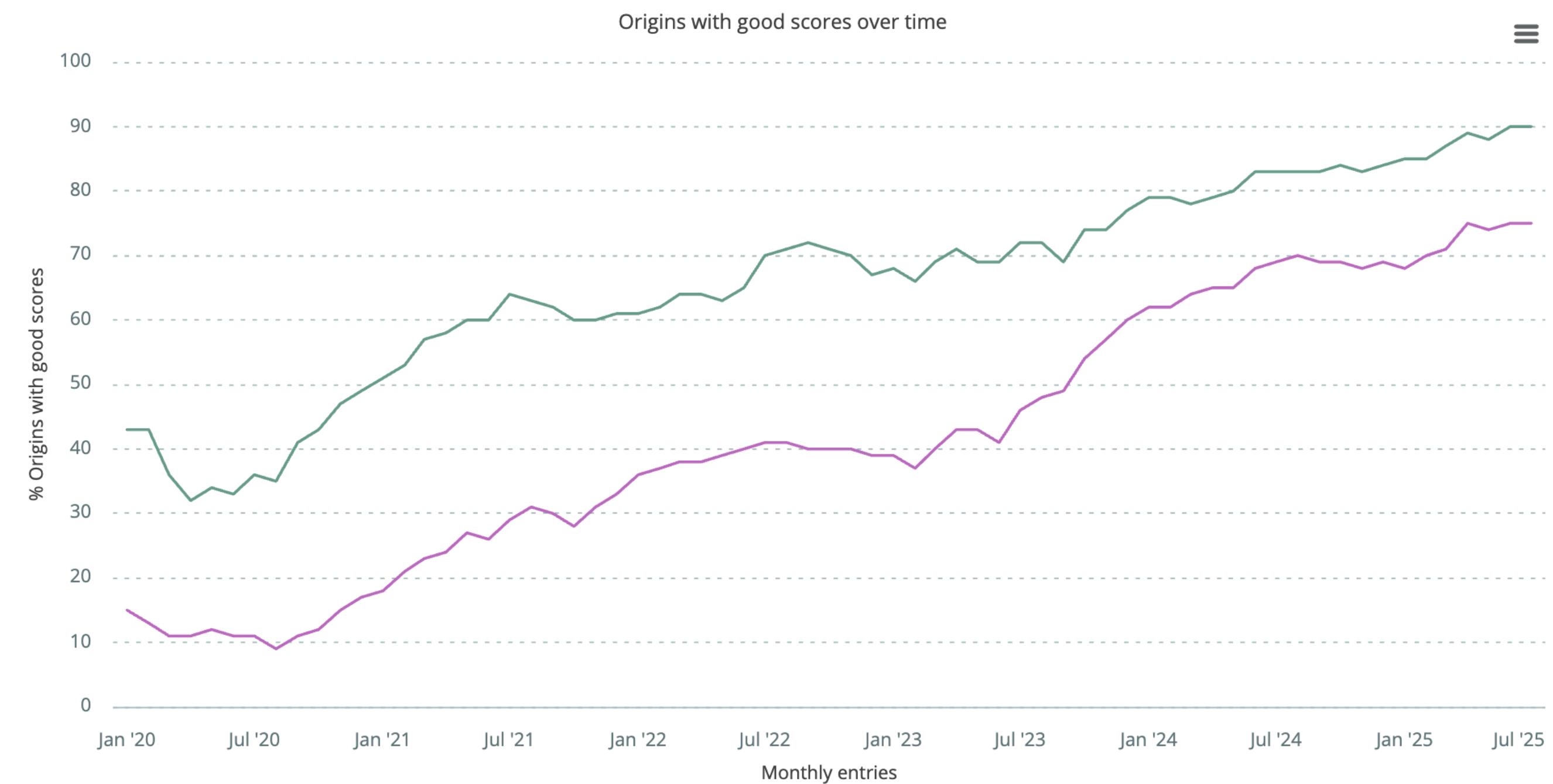
Good Largest Contentful Paint

Largest Contentful Paint (LCP) is an important, stable Core Web Vital metric for measuring perceived load speed because it marks the point in the page load timeline when the page's main content has likely loaded—a fast LCP helps reassure the user that the page is useful. Good experiences are less than or equal to 2.5 seconds.

LATEST DATA: 2025-07-01

Desktop — 90%
– 0.00%

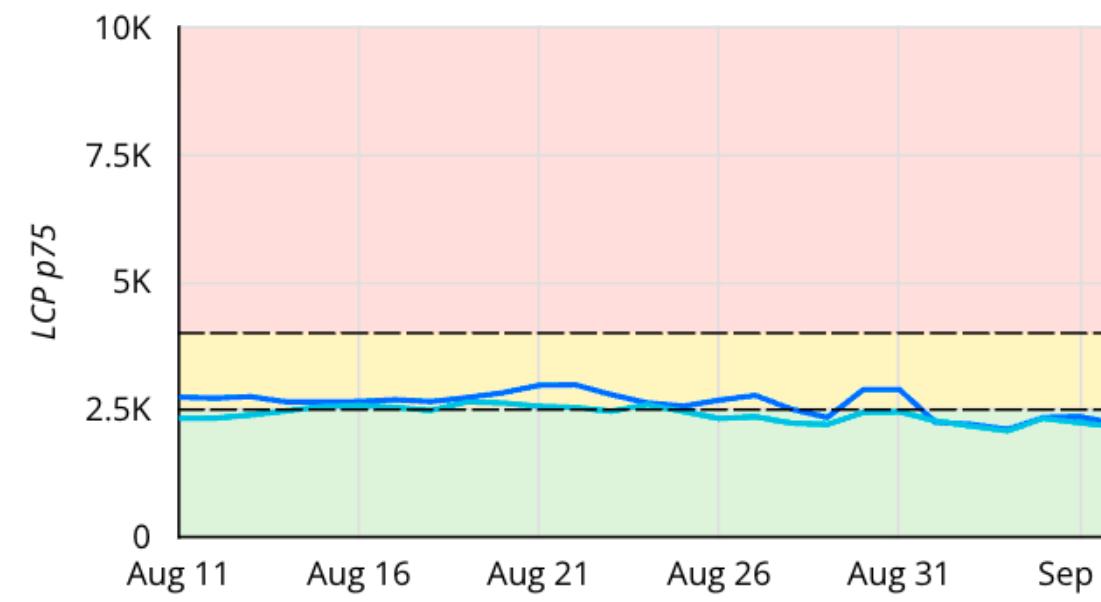
Mobile — 75%
– 0.00%



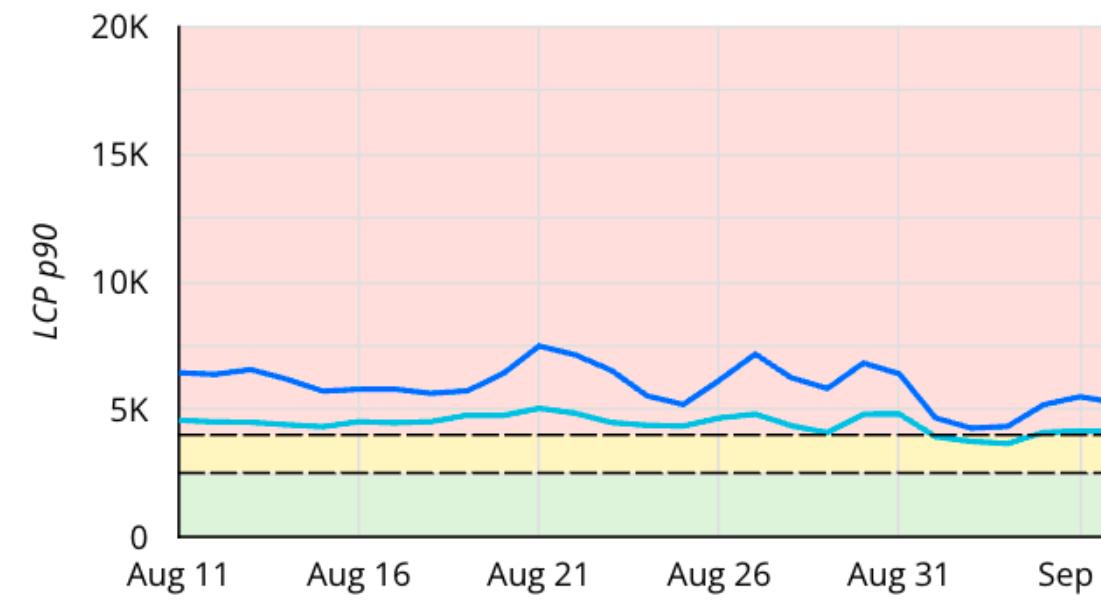
RUM @ Squarespace

LCP by country

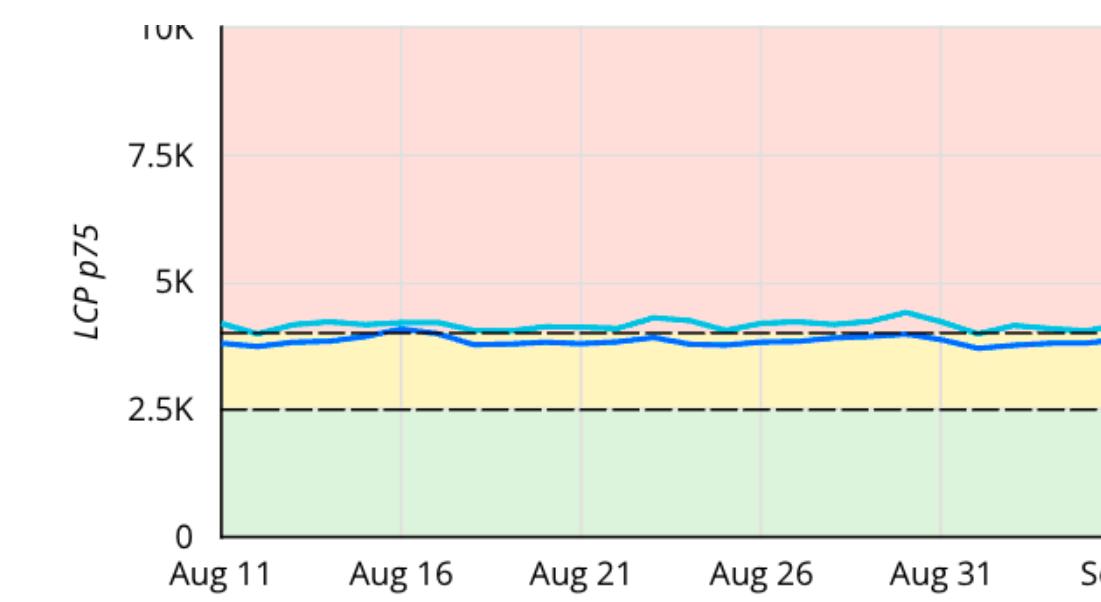
p75



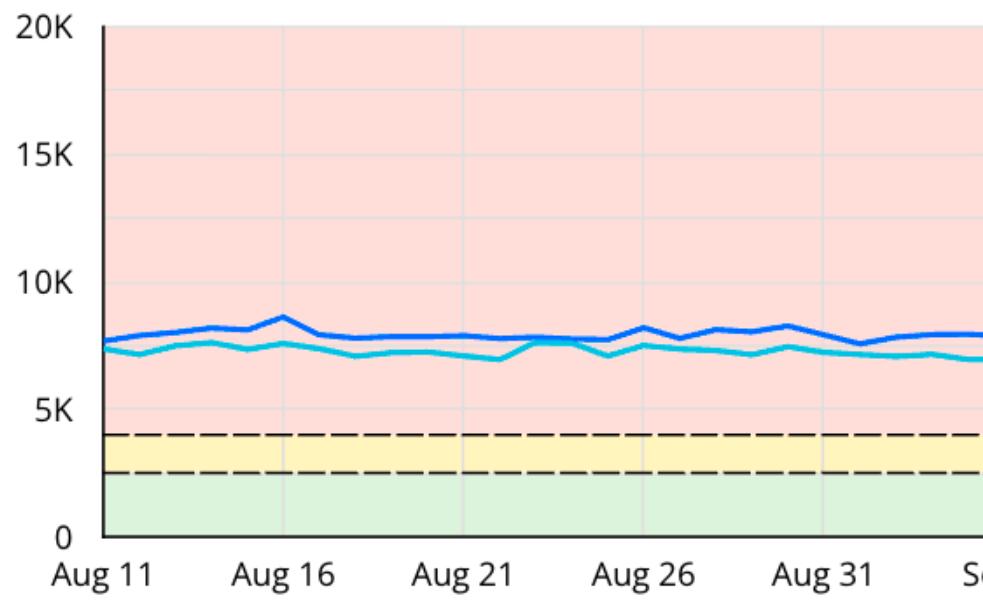
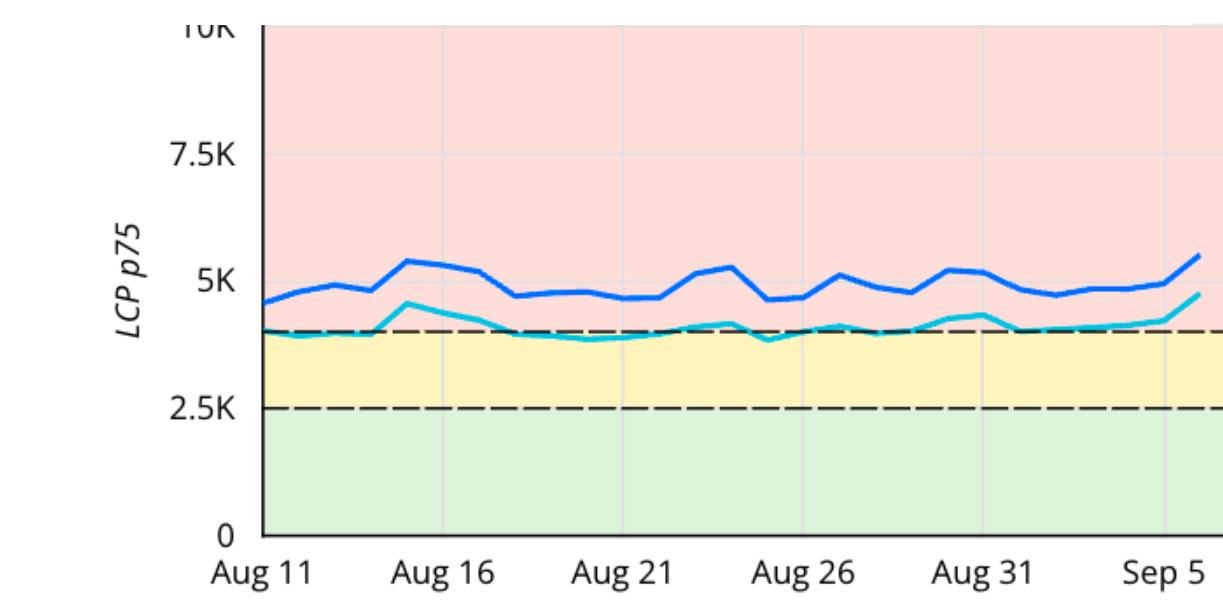
p90



Argentina



India





DECEMBER 21, 2012

Page Weight Matters

Three years ago, while I was a web developer at YouTube, one of the senior engineers began a rant about the page weight of the video watch page being far too large. The page had ballooned to as high as 1.2MB and dozens of requests. This engineer openly vented that “if they can write an entire Quake clone in under 100KB, we have no excuse for this!” Given that I agreed with him and I was excited to find a new project, I decided to champion the cause of getting the YouTube watch page to weigh in under 100KB. On the shuttle home from San Bruno that night, I coded up a prototype. I decided to limit the functionality to just a basic masthead, the video player, five related videos, a sharing button, a flagging tool, and ten comments loaded in via AJAX. I code-named the project “Feather”.

Even with such a limited set of features, the page was weighing in at 250KB. I dug into the code and realized that our optimization tools (i.e. Closure compilation) were unable to exclude code that was never actually used in

12,752
KUDOS

<https://blog.chriszacharias.com/page-weight-matters>

so, web components?

[Getting started](#)[Dev mode](#)[Registry conflicts](#)[Components](#)[Accordion](#)[Action Bar](#)[Action Button](#)[Action Group](#)[Action Menu](#)[Alert Dialog](#)[Asset](#)[Avatar](#)[Badge](#)[Banner](#)[Button](#)[Button Group](#)[Card](#)[Checkbox](#)[Coachmark](#)[Color Area](#)[Color Field](#)[Color Handle](#)

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.

[View on GitHub](#)

Accessible by default

Developed with existing and emerging browser specifications in mind, our components deliver a high quality experience whether using a screen reader, keyboard navigation, and/or customized contrast.

Standards based

[Web Components](#) are a set of technologies that work together, letting you create custom elements that work just like the standard HTML elements built into your browser.

Light weight

Spectrum Web Components are implemented using the [LitElement](#) base class. LitElement is designed for creating web components with a minimum amount of overhead.

Framework agnostic

You can use web components with any framework, since they're supported and encapsulated at the browser level.

Sample element usage

[Use Spectrum Web Component buttons](#)

```
<sp-button variant="accent" href="components/button">
```

```
  Use Spectrum Web Component buttons
```

Ps

← → C https://photoshop.adobe.com/id/tmp:rct9zw?dpasset=https%3A%2F%2Fhelpx.adobe.com%2Fcontent%2Fdam%2Fhelp%2Fen%2Fphotoshop%2Fhow-to%2Fin-app%2Fget-started-photoshop-web... ☆ | Finish update

Ps Get Started

Upgrade Download

121%

Size & position X

Move

Auto-select Layer

Align

Advanced settings

Transform

Crop

WHAT

A screenshot of the Adobe Photoshop Web Edition interface. On the left, a sidebar lists various tools: Move, Auto-select (Layer selected), Align, Advanced settings, Transform, Crop, and others. The main workspace shows a landscape photograph of a waterfall under a sky with a large, bright moon and scattered clouds. A text layer with the word "WHAT" in bold black capital letters is selected in the foreground. The "Move" tool is active, as shown by the blue selection border around the text and the cursor icon. The Photoshop logo is visible in the top left corner, and the top right features standard browser controls like upgrade, download, and search.

→ C https://photoshop.adobe.com/id?dpasset=https%3A%2F%2Fhelpx.adobe.com%2Fcontent%2Fdam%2Fhelp%2Fen%2Fphotoshop%2Fhow-to%2Fin-app%2Fget-started-photoshop-web%2F

Elements Console Sources Network >

✖ 45 ⚠ 3 ! 1 ⚙ : X

```
<div id="root"> slot
<!-->
<psw-app>
  #shadow-root (open)
    <!-->
    <!--?lit$171417826$-->
  <psw-app-context>
    #shadow-root (open)
      <psw-video-surface caption-language> slot
        #shadow-root (open)
          <ue-drawer width="290" dir="ltr" justify="end" animate open> flex slot
            #shadow-root (open)
              <div id="appView" slot="main"> flex slot
                <!--?lit$171417826$-->
                <psw-app-navbar> ... </psw-app-navbar>
                <!--?lit$171417826$-->
              <psw-document-page> flex
                #shadow-root (open)
                  <!-->
                  <!--?lit$171417826$-->
                <psw-doc> flex
                  #shadow-root (open)
                    <!-->
                    <!--?lit$171417826$-->
                  <psw-full-toolbar showing-panel> flex
                    #shadow-root (open)
                      <!-->
                      <psw-left-taskbar can-place sectionid="size-position" toolid="16" ff" extended> flex
                        #shadow-root (open)
                          <!-->
                          <div class="section-toolbar"> flex
                            <sp-action-button id="section-toolbar-size-position" data-dp ze & position selected dir="ltr" tabindex="0" role="button" emphasized> ... </sp-action-button> flex
                            <!--?lit$171417826$-->
                            <sp-action-button id="section-toolbar-adjust" data-dp-id="ad tabindex="0" role="button" aria-label="Adjust"> ... </sp-action-button>
                            <sp-action-button id="section-toolbar-select" data-dp-id="se r" tabindex="0" role="button" aria-label="Select"> ... </sp-action-button>
                            <sp-action-button id="section-toolbar-retouch" data-dp-id="s "ltr" tabindex="0" role="button" aria-label="Retouch"> ... </sp-action-button>
```

Ps | Get Started ⌂ 67% Get desktop app Download ⌂ ?

Size & position

Adjust

Select

Retouch

Quick actions

Paint

Shapes

Type

Edit background

Add image

Eyedropper

Foreground color

Switch

Background color

Collapse panel

A photograph of a waterfall cascading over rocks into a pool of water. A large, semi-transparent white circle is drawn over the middle of the waterfall, likely representing a selection or a mask in Photoshop. The sky is blue with some white clouds.

Documentation

< All Technologies

Apple Pay on the Web

Essentials

Using the Apple Pay JS SDK for iOS 18

Apple Pay setup

Setting Up Your Server

Configuring Your Environment

Maintaining Your Environment

Apple Pay Later visual merchandising widget

Adding an Apple Pay Later visual merchandising widget

Apple order tracking button

Adding a Track with Apple Wallet button

Apple Pay buttons

Displaying Apple Pay Buttons Using JavaScript

> ApplePayButton

> Displaying Apple Pay Buttons Using CSS

Apple Pay JavaScript APIs

Choosing an API for Implementing Apple Pay on Your W...

> Apple Pay on the web version history

> Apple Pay JS API

> Payment Request API

Reference

Filter

[Apple Pay on the Web](#) / [Displaying Apple Pay Buttons Using JavaScript](#)

Article

Displaying Apple Pay Buttons Using JavaScript

Load and configure the JavaScript Apple Pay button.

Overview

The JavaScript Apple Pay button provides a variety of Apple Pay button types that you can use on your website to initiate a transaction. You can specify the Apple Pay button style, type, and localization using attributes. Use CSS to set other properties, such as the size and corner radius.

Use the following code to load the button script into your webpage from the content delivery network:

```
<script src="https://applepay.cdn-apple.com/jsapi/v1.1.0/apple-pay-sdk.js"></script>
```

Display the Button

Add an `apple-pay-button` element to your webpage to display the button. Set the style, type, and locale when you add the button. The following code adds a black Buy with Apple Pay button localized to Greek, using the default size and corner radius:

```
<apple-pay-button buttonstyle="black" type="buy" locale="el-GR"></apple-pay-button>
```

Use CSS to customize the size and other attributes. The following example adds the same Greek button with a height of 40 pixels, a width of 140 pixels, and a corner radius of 5 pixels. The height includes the top and bottom padding.



The Firefox UI is now built with Web Components

November 2019

A couple of weeks ago, we landed a [commit](#) that took years of effort at Mozilla. It removed “XBL”, which means we’ve completed the process of migrating the Firefox UI to Web Components. It wasn’t easy – but I’ll get to that later.

The Firefox UI can be thought of as a very large single page app. It was built with DOM and JS from the start, which was a bold technology choice for a native application 20+ years ago. Because of this, Mozilla implemented some features necessary to build complex web applications way before the Web platform supported them.

Over time, these features have [evolved](#) into specifications like CSS flexbox and Web Components. When this happens, it’s easy to allow the existing codebase to continue to use the original version, and require the platform supports both features. This makes sense – rewriting legacy code that already works is difficult and not always cost effective.

But we made a decision while Web Components were being implemented in Firefox to start a parallel project in which we would migrate our existing UI components to use them. We decided to do this in an incremental way, so that each individual change could land while keeping Firefox in a functional state, as opposed to branching and rewriting the UI from scratch.

It's taken a couple weeks, but <https://briangrinstead.com/blog/firefox-webcomponents/> along with the support of the rest of the organization, and I'm happy to report that we've now finished. This is a big

0 Tags

Q Go to file

t

+

<> Code ▾

About

ME.md

7183a93 · 10 months ago

12 Commits

Initial commit

10 months ago

relative-time.sc-aXZVg 95.04 × 16.67

eadme

Initial commit

10 months ago

Update README.md

10 months ago

MIT license

Activity

sole

Sources

Network

Performance

Memory

Application

>>

✖ 7

▼ <div class="react-directory-commit-age">
 ▼ <relative-time class="sc-aXZVg" tense="past" datetime="2023-12-11T20:50:38.000Z" title="Dec 11, 2023, 3:50 PM EST"> == \$0

► #shadow-root (open)

"Dec 11, 2023"

</relative-time>

</div>

Filter

HTML

[> Guides](#)

How to

[Define terms with HTML](#)[Use data attributes](#)[Use cross-origin images](#)[Add a bitmap on top of an image](#)[Author fast-loading HTML pages](#)[Add JavaScript](#)

Reference

[> Elements](#)[> Attributes](#)[> Global attributes](#)[> Attributes by element](#)[> Attribute values](#)

HTML: HyperText Markup Language

HTML (HyperText Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation ([CSS](#)) or functionality/behavior ([JavaScript](#)).

"Hypertext" refers to links that connect web pages to one another, either within a single website or between websites. Links are a fundamental aspect of the Web. By uploading content to the Internet and linking it to pages created by other people, you become an active participant in the World Wide Web.

HTML uses "markup" to annotate text, images, and other content for display in a Web browser. HTML markup includes special "elements" such as `<head>`, `<title>`, `<body>`, `<header>`, `<footer>`, `<article>`, `<section>`, `<p>`, `<div>`, ``, ``, `<aside>`, `<audio>`, `<canvas>`, `<datalist>`, `<details>`, `<embed>`, `<nav>`, `<search>`, `<output>`, `<progress>`, `<video>`, ``, ``, `` and many others.

An HTML element is set off from other text in a document by "tags", which consist of the element name surrounded by `<` and `>`. The name of an element inside a tag is case-insensitive. That is, it can be written in uppercase, lowercase, or a mixture. For example, the `<title>` tag can be written as `<Title>`, `<TITLE>`, or in any other way. However, the convention and recommended practice is to write tags in lowercase.

The articles below can help you learn more about HTML.

Beginner's tutorials

Our [learn web development core modules](#) contain modern, up-to-date tutorials covering HTML fundamentals.

In this article

Beginner's tutorials

[Guides](#)[Reference](#)[Related topics](#)

AD

imagekit.io

Image & Video API
Real-time URL transforms, global CDN delivery & easy integration with existing storage.

[Start Free](#)

[Don't want to see ads?](#)

the components of... web components



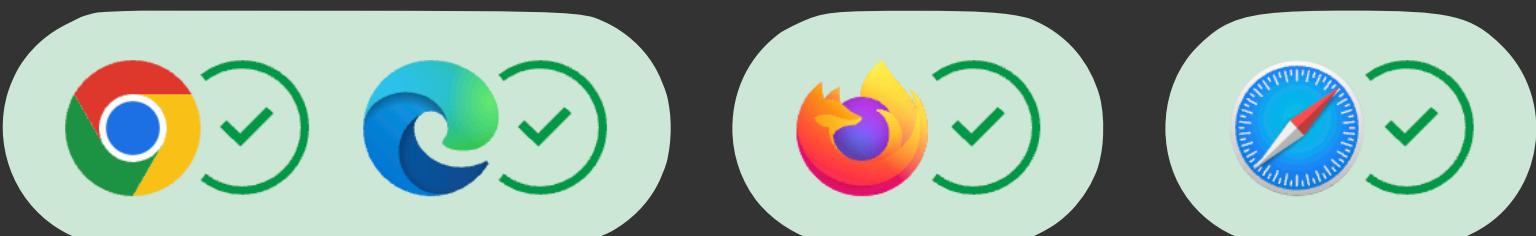
Custom Elements



HTML Templates



Shadow DOM & Slots



Custom elements

A Custom Element

```
<my-element>
```

Hey, I'm a custom element!

```
</my-element>
```

nothing to see here...

```
<x-->
```

Completely valid custom element.

```
</x-->
```

all valid! I don't make the rules.

<my-card></my-card>

<what-the-heck></what-the-heck>

<a-></a->

<a-4-_></a-4-_>

<card-10.0.1></card-10.0.1>

<hi-😍></hi-😍>

using custom elements

```
<p>I'm a paragraph</p>
```

```
<my-element>I'm a custom element!</my-element>
```

```
<ul>
```

```
  <li>I'm a list item</li>
```

using custom elements

```
<my-element>  
  <p>I'm a paragraph in a custom element!</p>  
</my-element>
```

using custom elements

```
<my-element></my-element>
```

html rules apply!

Nope.

```
<ul>  
  <my-list-item>I can't go here.</my-list-item>  
  <my-list-item>I can't go here.</my-list-item>  
</ul>
```

Okay.

```
<ul>  
  <li><my-list-item>I can go here!</my-list-item></li>  
</ul>
```

The lifecycle

Defining an element

```
<my-element>My Element</my-element>
```

```
<script>
```

```
customElements.define("my-element", class extends HTMLElement {});
```

```
</script>
```

lifecycle moments

```
class MyElement extends HTMLElement {  
  constructor(){}
  connectedCallback(){}
  attributeChangedCallback(){}
  disconnectedCallback(){}
  adoptedCallback(){}
}  
customElements.define("my-element", MyElement);
```

connected Callback: your new DOM-ready

```
<my-element>Hey, I'm a custom element!</my-element>

<script>
customElements.define('my-element', class extends HTMLElement {
  connectedCallback(){
    console.log('my-element is in the DOM & ready for enhancements!');
  }
});

</script>
```

connectedCallback: More practically

```
<my-element><button>My Element</button></my-element>

<script>
customElements.define("my-element", class extends HTMLElement {
  connectedCallback(){
    this.btn = this.firstElementChild;
    this.btn.addEventListener("click", e => { this.handleClick(e); } );
  }
  handleClick(e){
    console.log(this, e, e.target);
  }
});
</script>
```

lifecycle moments

```
class MyElement extends HTMLElement {  
  constructor(){}
  connectedCallback(){}
  attributeChangedCallback(){}
  disconnectedCallback(){}
  adoptedCallback(){}
}  
customElements.define("my-element", MyElement);
```

HTML Templates

A template element

```
<template>  
  <li class="user">  
    <strong class="name"></strong>  
    <span class="bio"></span>  
  </li>  
</template>
```

Shadow DOM & Slots

Preferences

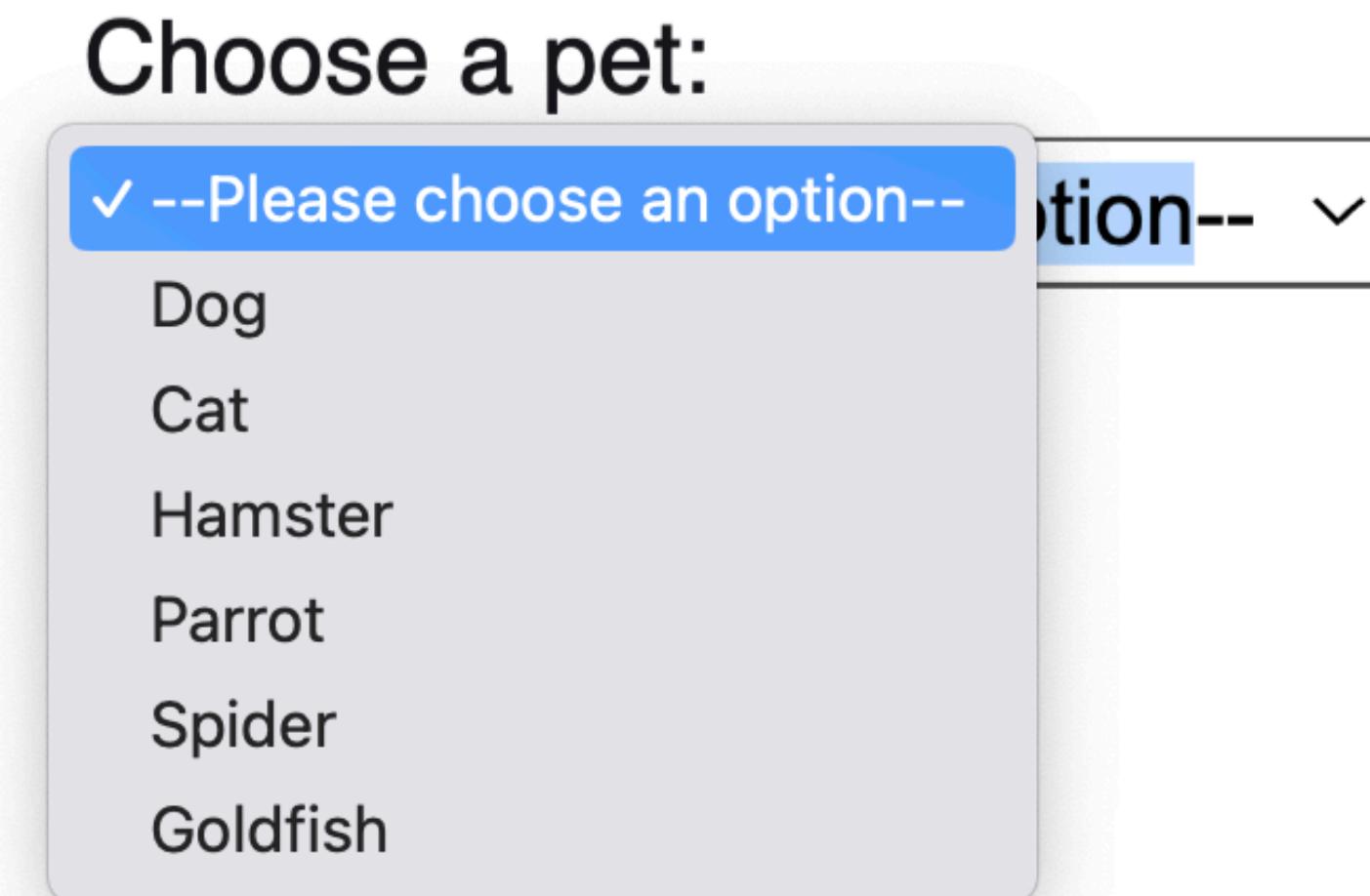
Elements

Show user agent shadow DOM

Word wrap

Show HTML comments

```
1 <label for="pet-select">Choose a pet:</label>
2
3 <select name="pets" id="pet-select">
4   <option value="">--Please choose an option--
5   <option value="dog">Dog</option>
6   <option value="cat">Cat</option>
7   <option value="hamster">Hamster</option>
8   <option value="parrot">Parrot</option>
9   <option value="spider">Spider</option>
10  <option value="goldfish">Goldfish</option>
11 </select>
12
```



Elements Console Sources Network Performance Memory Application

▼ <body>

 ▼ <div id="html-output" class="output editor-tabbed">

 <label for="pet-select">Choose a pet:</label>

 ▼ <select name="pets" id="pet-select"> == \$0

 ▼ #shadow-root (user-agent)

 <div aria-hidden="true">--Please choose an option--</div>

 ▼ <slot id="select-options">

 <div>--Please choose an option--</div>

 ▼ <slot id="select-options">

We can use it too!

element ensures your site displays the latest style, renders correctly across devices, and follows Apple guidelines. For design guidance, see [Human Interface Guidelines > Apple Pay > Buttons and Marks](#).

Try it: Display settings

Use the following tools to try the different display settings on the button below:

BUTTON STYLE	BUTTON TYPE	BUTTON LANGUAGE
Black (De... ▼	Plain ▼	English (... ▼

BUTTON WIDTH	BUTTON HEIGHT	BUTTON CORNER RADIUS
<input type="range"/> Width: 179px	<input type="range"/> Height: 35px	<input type="range"/> Corner Radius: 3px

BUTTON PADDING Y	BUTTON PADDING X	BUTTON BOX SIZING
<input type="range"/> Button Padding Y: 0px	<input type="range"/> Button Padding X: 0px	Border Box ▼

```
<style>
apple-pay-button {
  --apple-pay-button-width: 179px;
  --apple-pay-button-height: 35px;
  --apple-pay-button-color: black;
}
```



k;">

▼<apple-pay-button buttonstyle="black" type="plain" locale="en-US" onclick="onApplePayButtonClicked()" lang="en-US" style="--apple-pay-button-height: 35px; --apple-pay-button-width: 179px;">

▼#shadow-root (open)

... ▶<style>...</style> == \$0

▼<div class="apple-pay-btn-wrapper en-US" style="visibility: visible;">

▼<button aria-label="Apple Pay" class="apple-pay-btn black plain">

▼ flex

▼<svg id="svg-logo" viewBox="0 0 35 15">

<!--?xml version="1.0" encoding="UTF-8"?-->

▼<svg id="svg-logo" x="0" y="3.85" height="7.3" width="35" class="logo" viewBox="0 0 105 43" version="1.1" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">

<title>Apple Logo</title>

▼<g stroke="none" stroke-width="1" fill="none" fill-rule="evenodd">

▼<g fill="#FFF">

<path d="M19.4028,5.5674 C20.6008,4.0684 21.4138,2.0564 21.1998,0.0004 C19.4458,0.0874 17.3058,1.1574 16.0668,2.6564 C14.9538,3.9414 13.9688,6.0374 14.2258,8.074 C16.1948,8.1784 18.1618,7.0244 19.4028,5.5674"></path>

<path d="M21.1772,8.3926 C18.3182,8.2226 15.8872,10.0156 14.5212,10.0156 C13.1552,10.0156 11.0642,8.4786 8.8022,8.5196 C5.8592,8.5626 3.1282,10.2276 1.6342,12.8746 C-1.4378,18.1696 0.8232,26.0246 3.8122,30.3376 C5.2622,32.4716 7.0102,34.8206 9.3142,34.7366 C11.4912,34.6506 12.344

div.applePayButtonContainer apple-pay-button #shadow-root style

Styles Computed Layout Event Listeners DOM Breakpoints >>

Back to that template...

```
<template>
```

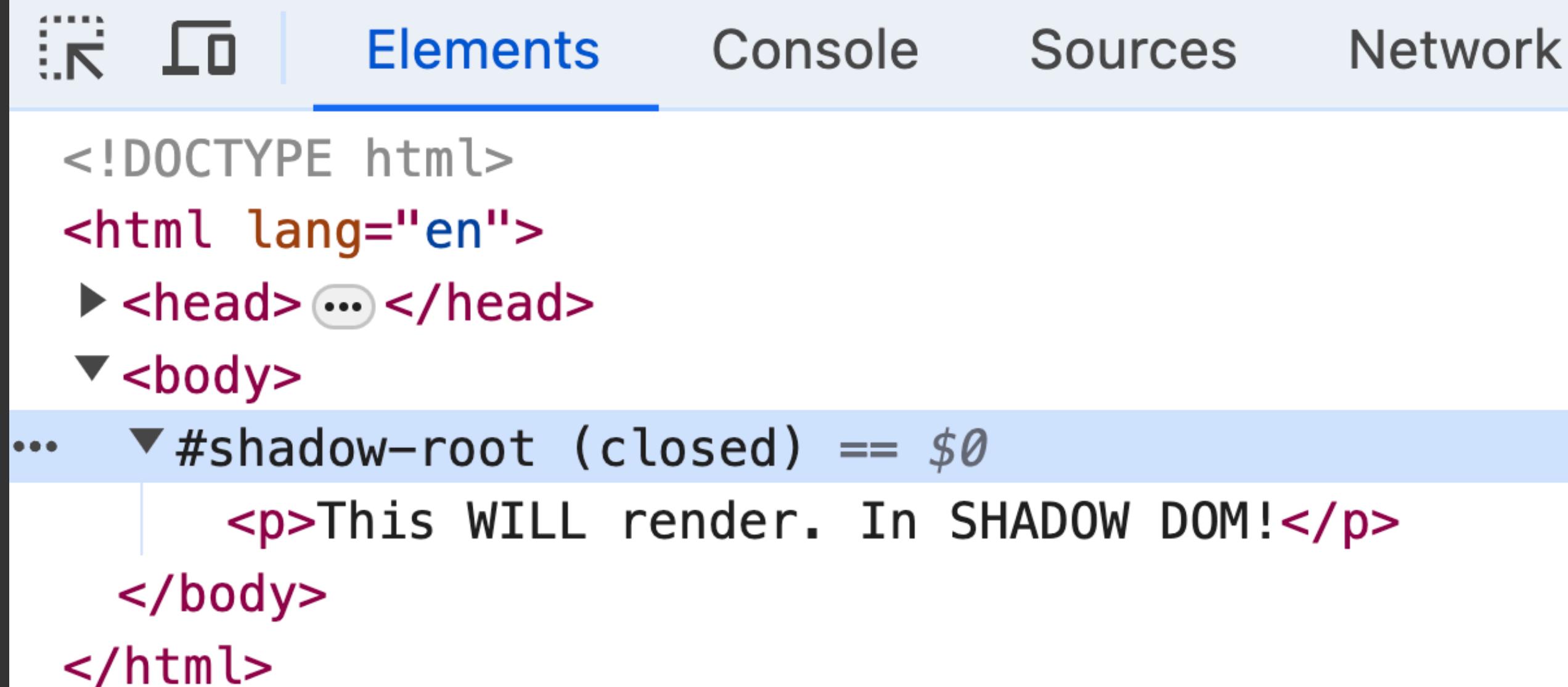
```
  <p>This won't render. It's meant for devs</p>
```

```
</template>
```

templates can ALSO make shadow dom

```
<template shadowrootmode="closed">  
  <p>This WILL render.  
    In SHADOW DOM!</p>  
</template>
```

This WILL render. In SHADOW DOM!



The screenshot shows the Chrome DevTools interface with the 'Elements' tab selected. The page source is visible at the bottom, showing the rendered HTML structure. The rendered content includes the DOCTYPE, HTML, head, body, and a shadow-root element. Inside the shadow-root, there is a p element containing the text "This WILL render. In SHADOW DOM!". The shadow-root is identified by the ID '#shadow-root (closed) == \$0'. The rest of the page source shows the standard HTML structure above and below the shadow-root.

```
<!DOCTYPE html>
<html lang="en">
  > <head> ...
  > <body>
...
  >   > #shadow-root (closed) == $0
        >     <p>This WILL render. In SHADOW DOM!</p>
      >   </body>
    > </html>
```

Alternatively, add a shadow with js

```
<my-element></my-element>

<script>
customElements.define('my-element', class extends HTMLElement {
  constructor(){
    super();
    this.attachShadow({mode: "closed"});
    this.shadowRoot.innerHTML = "<p>Hi I'm in shadow DOM</p>";
  }
}) ;

</script>
```

```
<!DOCTYPE html>
<html lang="en">
  > <head> ...
  >   </head>
  > <body>
  ... >   > #shadow-root (closed) == $0
  ... >     <p>This WILL render. In SHADOW DOM!</p>
  ... >   </body>
  ... > </html>
```

html body #shadow-root

⋮ [Console](#) [What's new](#) [Issues](#)

▶ ⚡ top ▾ ⚡ [Filter](#)

```
> document.querySelector("p")
< null
```

Open shadow root mode

```
<div>  
  <template shadowrootmode="open">  
    <p>This WILL render. In SHADOW DOM!</p>  
  </template>  
</div>
```

```
<!DOCTYPE html>
<html lang="en">
  > <head> ... </head>
  > <body>
...   > <div> == $0
      > #shadow-root (open)
          > <p>This WILL render. In SHADOW DOM!</p>
      > </div>
    > </body>
  > </html>
```

html body div

⋮ Console What's new Issues

▶ top ▾ ⚡ Filter

> document.querySelector("div").shadowRoot

< > #shadow-root (open)
 > <p>This WILL render. In SHADOW DOM!</p>

How about a slot instead

```
<my-element><p>Hi I'm in shadow DOM</p></my-element>

<script>
customElements.define('my-element', class extends HTMLElement {
  constructor(){
    super();
    this.attachShadow({mode: "open"});
    this.shadowRoot.innerHTML = "<slot></slot>";
  }
}) ;

</script>
```

How about a slot instead

```
<my-element>Mirror</my-element>
```

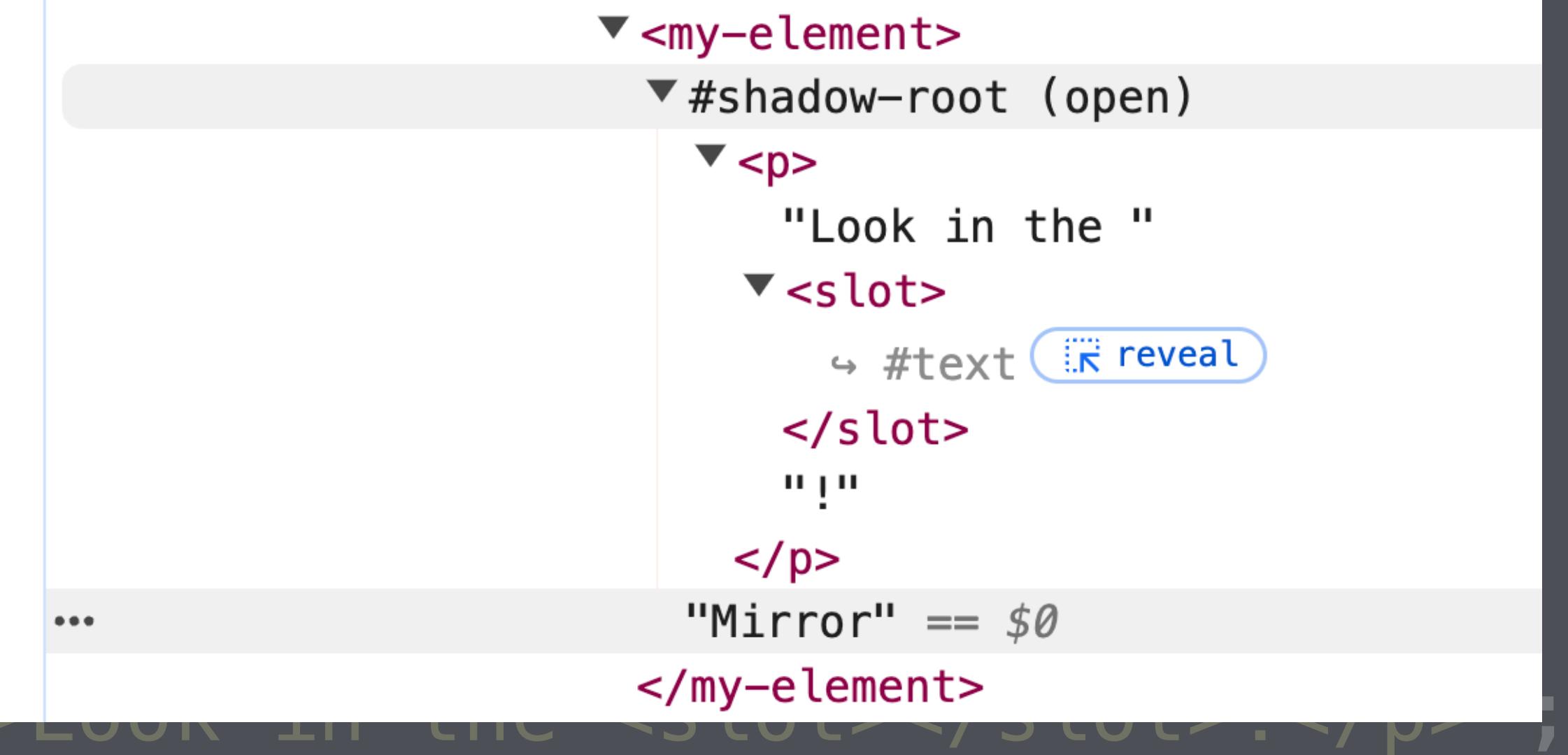
```
<script>
```

```
Look in the Mirror!
```

```
}
```

```
} );
```

```
</script>
```



css encapsulation...

```
<style>  
p {  
color: red;  
}  
</style>
```

```
<p>Hi</p>  
<div>  
  <template shadowrootmode="open">  
    <p>Hi</p>  
  </template>  
</div>  
<p>Hi</p>
```

Hi

Hi

Hi

what about body?

```
<style>
```

```
body {
```

```
color: red;
```

```
}
```

```
</style>
```

```
<p>Hi</p>
```

```
<div>
```

```
  <template shadowrootmode="open">
```

```
    <p>Hi</p>
```

```
  </template>
```

```
</div>
```

```
<p>Hi</p>
```

Hi

Hi

Hi

fight!

```
<style>
body {
color: red;
}
</style>

<p>Hi</p>
<div>
  <template shadowrootmode="open">
    <style> p { color: blue; } </style>
    <p>Hi</p>
  </template>
</div>
<p>Hi</p>
```

Hi

Hi

Hi

```
<style>  
apple-pay-button {  
    --apple-pay-button-width: 148px;  
    --apple-pay-button-height: 44px;  
    --apple-pay-button-border-radius: 2px;  
    --apple-pay-button-padding: 0px 0px;  
    --apple-pay-button-box-sizing: border-box;  
}  
</style>  
<apple-pay-button buttonstyle="white" type="plain" locale=
```

can web components save us?



• Home Docs Showcase

Star On GitHub

★

31K

React For Cr

Highly customizable animated components that make your app look great.

Browse Components

 Web Awesome is here! Celebrate with 20% off on a Web Awesome Pro plan... for life!

 Get Pro + Save 20%



Your Projects

Docs



Get Pro

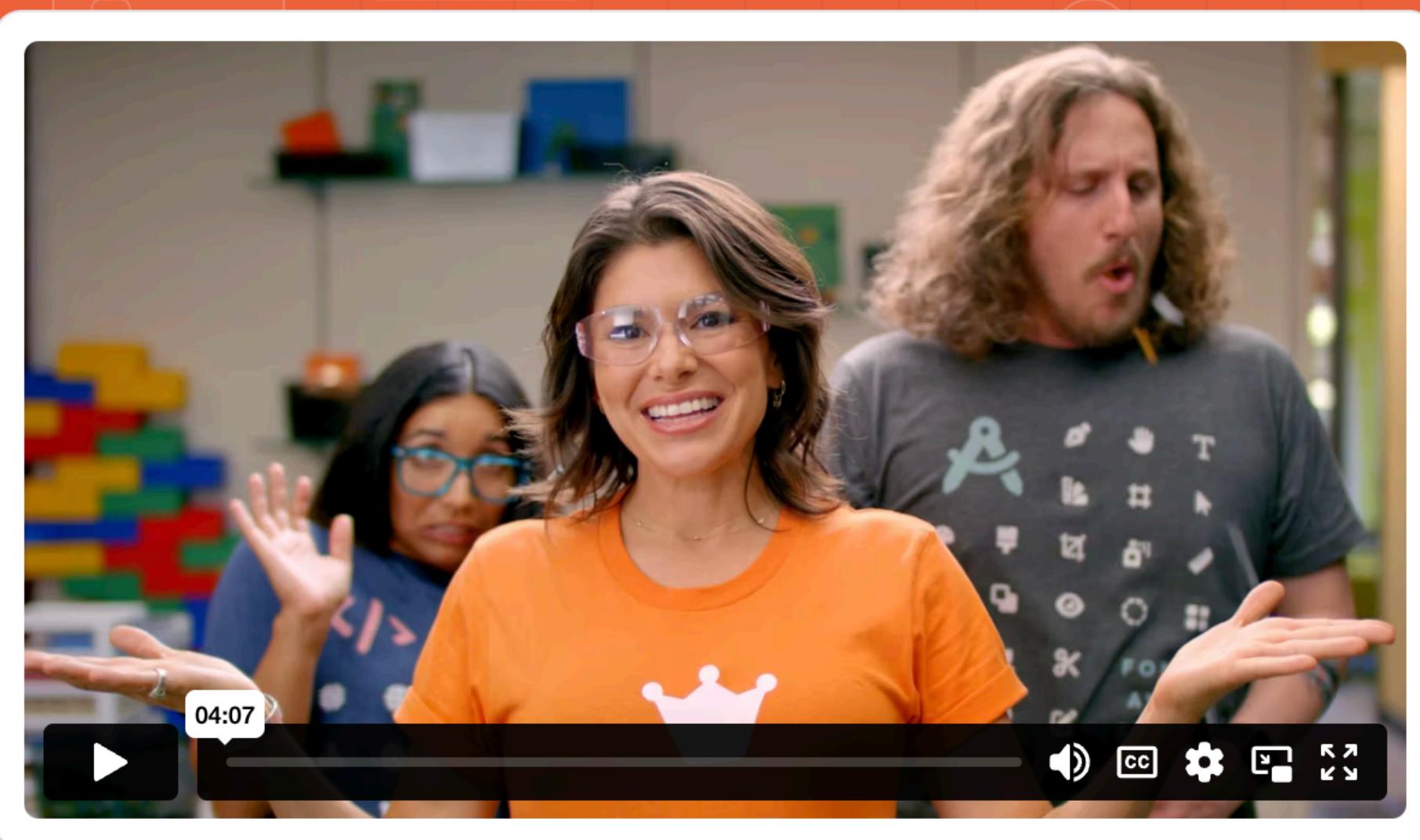


Build something awesome, standard-based, and uniquely yours!

Web Awesome is the biggest open-source library of meticulously designed, highly customizable, web-standards-based, and framework-agnostic UI components.

 Get Started

 Upgrade to Pro



v3.0.0

50+ Components

100+ Patterns

11 Themes

WEB STANDARDS-BASED

Readable, Durable, and Framework-Friendly

 Search the Docs for... /

Getting Started

Installation

Usage

Customizing

Form Controls

Localization

Resources

Help & Support

Source Code

Figma Design Kit

Pro

Accessibility

Browser Support

Contributing

Changelog

Visual Tests

Components

▼

Page 

Pro

Animated Image

Animation

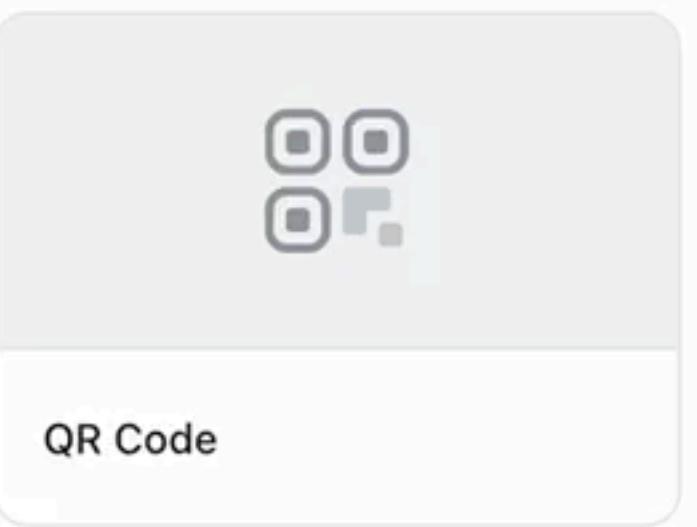
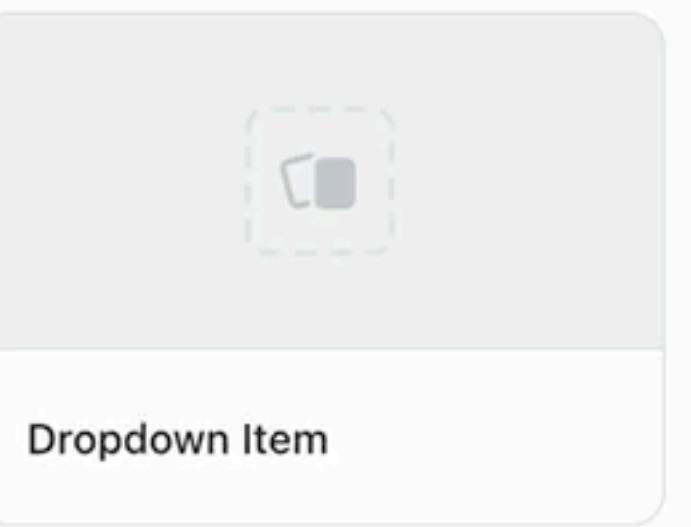
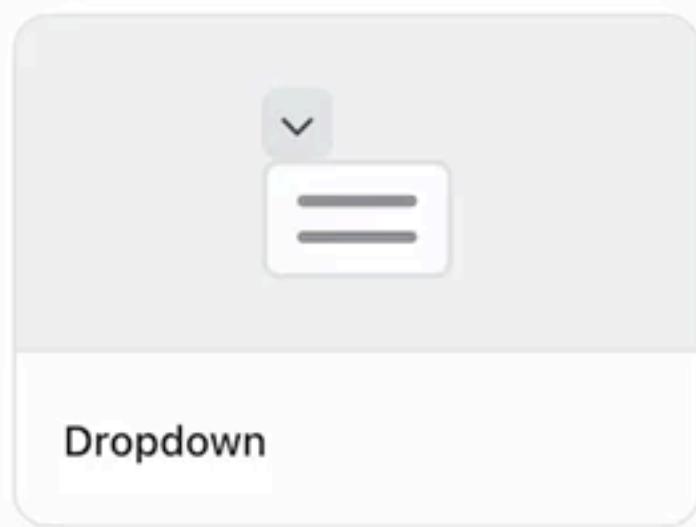
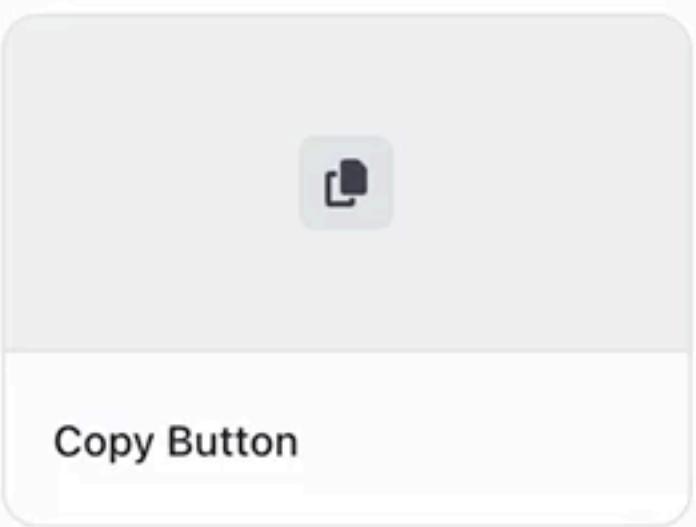
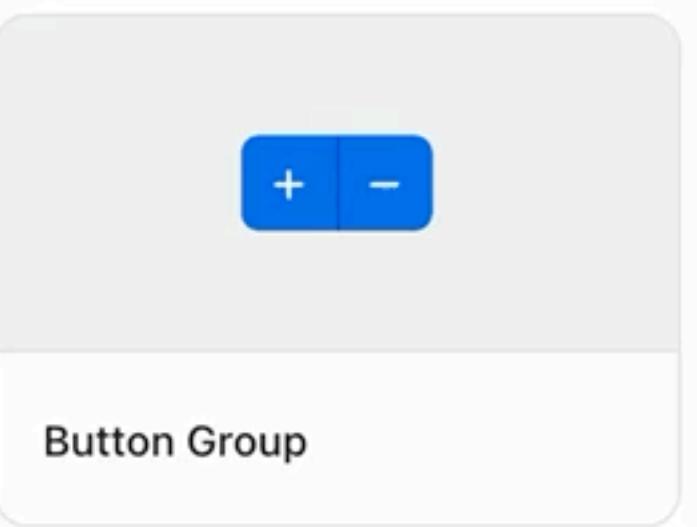
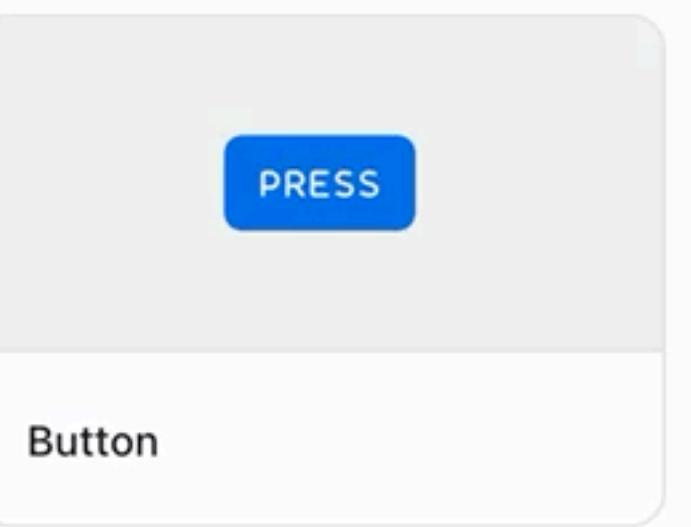
Avatar

Badge

Components

 Search Components

Actions



Components

Actions

Feedback & Status

Form Controls

Imagery

Navigation

Organization

Other

Utilities

Feedback & Status

<https://webawesome.com/>

If you're using a custom element to extend existing markup, that's an HTML web component.



Jeremy Keith

<https://adactio.com/journal/20618>

A QR Code Component

```
<q-r>
```

```
    <a href="https://scottjehl.com">My Website</a>
```

```
</q-r>
```

The Q-R Component Code

```
import { generate } from '/assets/q-r/leanqr.js';

customElements.define('q-r', class extends HTMLElement {

connectedCallback(){

    let destination = this.querySelector('a').href;
    const qrCode = generate(destination);

    this.canvas = document.createElement('canvas');
    this.canvas.setAttribute('role', 'img');
    this.canvas.setAttribute('aria-label', 'Scannable QR Code for the link that follows');

    qrCode.toCanvas(this.canvas);

    this.prepend(this.canvas);
}

});
```

you might not need
components



AJAX

[JSON](#)[Load](#)[Post](#)[Request](#)

Effects

[Fade In](#)[Fade Out](#)[Hide](#)[Show](#)[Toggle](#)

Elements

[Add Class](#)[After](#)[Append](#)[Append To](#)[Before](#)[Children](#)[Clone](#)[Closest](#)[Contains](#)[Contains Selector](#)[Contents](#)

You might not need jQuery

jQuery and its cousins are great, and by all means use them if it makes it easier to develop your application.

If you're developing a library on the other hand, please take a moment to consider if you actually need jQuery as a dependency. Maybe you can include a few lines of utility code, and forgo the requirement. If you're only targeting more modern browsers, you might not need anything more than what the browser ships with.

At the very least, make sure you know what [jQuery is doing for you](#), and what it's not.

Some developers believe that jQuery is protecting us from a great demon of browser incompatibility when, in truth, post-IE8, browsers are pretty easy to deal with on their own; and after the Internet Explorer era, the browsers do even more.

[★ Star on Github](#)

14,209

 Search...

Do you need to support IE?





Simple default popover

web.dev

PRO

+ Follow



Settings



/ HTML

⚙️ ⏺

* JS

```
1 <button popovertarget="my-popover"> Open Popover </button>
2
3 <div id="my-popover" popover>
4   <p>I am a popover with more information.</p>
5 </div>
6
```

Open Popover

 HTML *
CSS

```
1 ▼ <details name="tech-specs">
2 ▼   <summary>System configuration</summary>
3 ▼   <ul>
4 ▼     <li>200GB RAM</li>
5 ▼     <li>4PB storage</li>
6   </ul>
7 </details>
8 ▼ <details name="tech-specs">
9 ▼   <summary>Recommended settings</summary>
10 ▼   <ul>
11 ▼     <li>Extreme mode: on</li>
12 ▼     <li>Ravtracing: enabled</li>
```

- ▶ System configuration
- ▶ Recommended settings
- ▶ Other details

HTML

CSS

Result

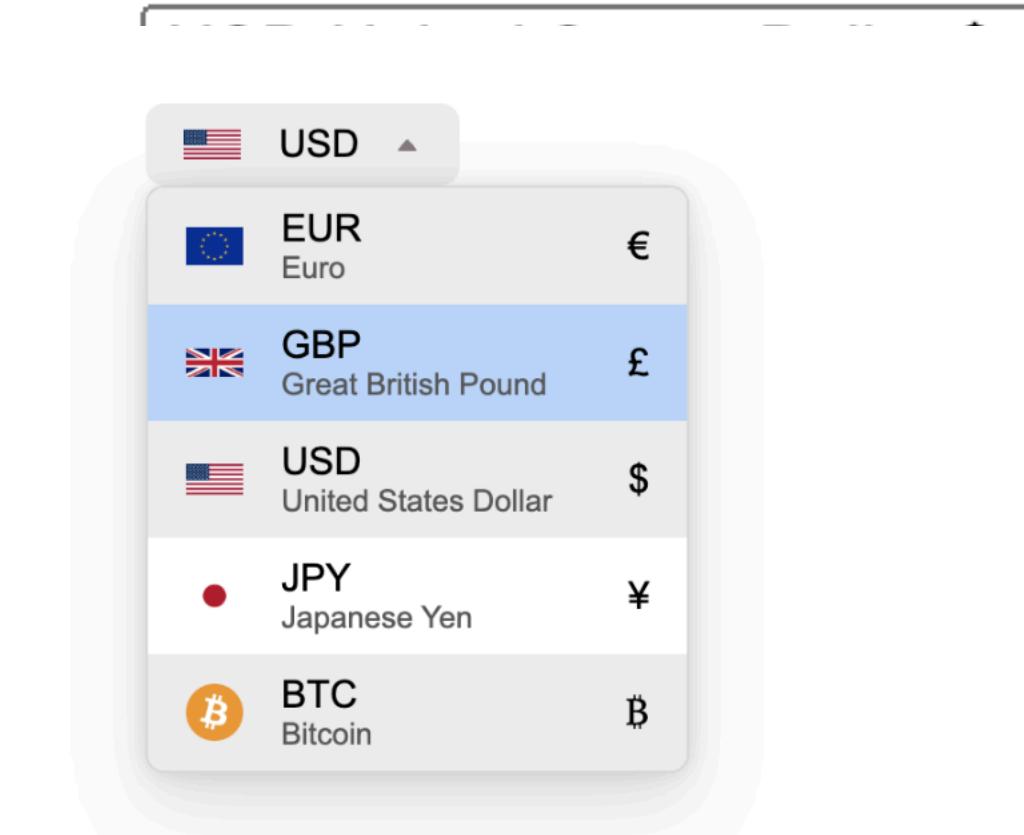
```
/* Enter into custom mode */

select,
::picker(select) {
  appearance: base-select;
}

/* remove checkmark */
option::before {
  display: none;
}

/* Styling the options list */

::picker(select) {
  border-radius: 0.5rem;
  padding: 0;
}
```



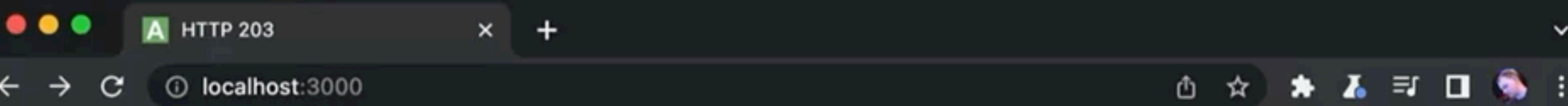
Resources

1x

0.5x

0.25x

Rerun



HTTP 203

The image displays a grid of six video thumbnail cards, each featuring two hosts (a man and a woman) against a dark background with a grid pattern. The thumbnails are arranged in two rows of three. Each card contains a timestamp at the bottom and a central text box with white text on a black background.

- Top Row:**
 - Thumbnail 1:** "Is .CSS a bad idea?"
HTTP 203
2022-05-03
 - Thumbnail 2:** Which key was pressed?
HTTP 203
2022-03-29
 - Thumbnail 3:** Building VR and AR experiences using HTML
HTTP 203
2022-03-15
- Bottom Row:**
 - Thumbnail 4:** Cross-fading DOM Elements
 - Thumbnail 5:** Generating your color palette in CSS
 - Thumbnail 6:** The ZOMBIE Dom



making the transition

Custom Elements Everywhere

Making sure frameworks and custom elements can be BFFs 🍻

What's this?

Custom Elements are the lynchpin in the Web Components specifications. They give developers the ability to define their own HTML elements. When coupled with Shadow DOM, Custom Elements should be able to work in any application. But things don't always work seamlessly.

This project runs a suite of tests against each framework to identify interoperability issues, and highlight potential fixes already implemented in other frameworks. If frameworks agree on how they will communicate with Custom Elements, it makes developers' lives easier.

<https://custom-elements-everywhere.com/>



Lit

make your own wc framework

```
<my-element><p>hi</p></my-element>
<script>
class BaseElement extends HTMLElement {
  $ = this.querySelector;
}
class myElement extends BaseElement {
  connectedCallback(){
    console.log( this.$('p') )
  }
}
customElements.define('my-element', MyElement);
</script>
```

A custom element base class

Published on December 19, 2023

I can't believe I've never tried making own my base class, considering that I've made my own [CSS reset](#).

I mentioned it in my post about [defining custom elements](#), so let's start with the static `register()` method.

```
class BaseElement extends HTMLElement {  
    static register(tagName = this.tagName) {  
        customElements.define(tagName, this);  
    }  
}
```

Now there's two boilerplatey parts that I find myself being tired of writing over and over:

1. Manipulating DOM elements
2. Attaching event handlers

For grabbing DOM references, `this.querySelector` feels more verbose than `document.querySelector`.

<https://mayank.co/blog/custom-element-base/>

**WEB COMPONENTS
CAN'T SAVE US.**

(alone)

**WEB COMPONENTS
CAN'T SAVE US.**

(alone)

But you can!

Thanks!

online course DISCOUNT!

Use code TWENTYFIVE to get the course for \$25

Visit <https://scottjehl.com>



Now Available!

WEB COMPONENTS *Demystified*

A COMPREHENSIVE, PREMIUM COURSE ABOUT BUILDING DYNAMIC, FAST,
RESILIENT APPS WITH STANDARD WEB COMPONENTS.

