

Adventures in Browser tooling



\$ whoami



- Chris Heilmann @codepo8 – he/him/his
- Principal Program Manager Microsoft Developer Experience (VSCode, Devtools...)
- W3C Member
- 20 years web developer
- eToys, Agilisys, Yahoo, Mozilla...
- Wrote 3 books, contributed to 10(?)...
- Blogged since 2004, loud on Twitter

Minding the gap...



A demands gap...

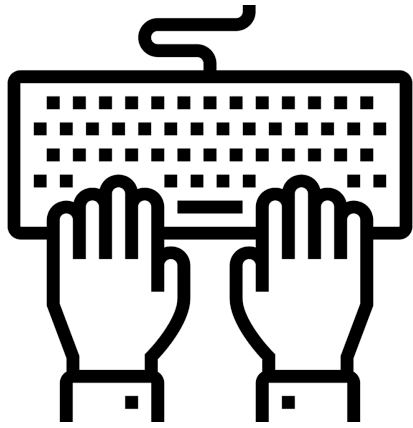
Useful, effective
and pretty amazing
things in devtools

What we use day
to day..

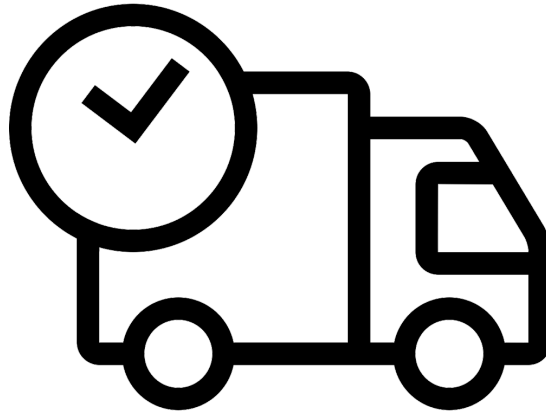


What we demand
of the web...

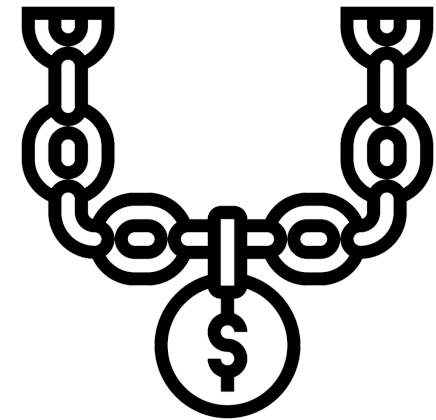
An assumptions gap...



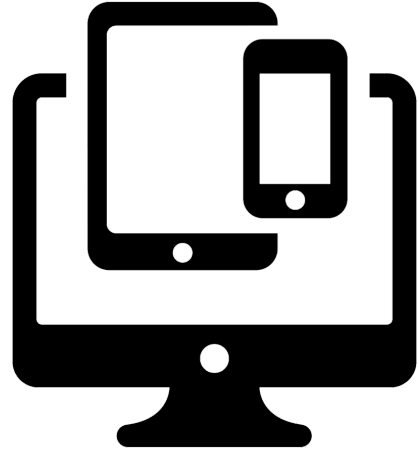
Build it!



Ship it!



Great success!



The web isn't a
native platform

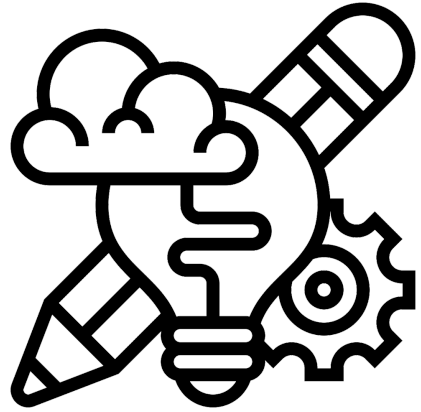
Our products should:

- work independent of device and operating system
- still cater to users of older devices and environments
- not make any assumptions
- be customizable by our users
- have a longer lifetime than the current state of technology

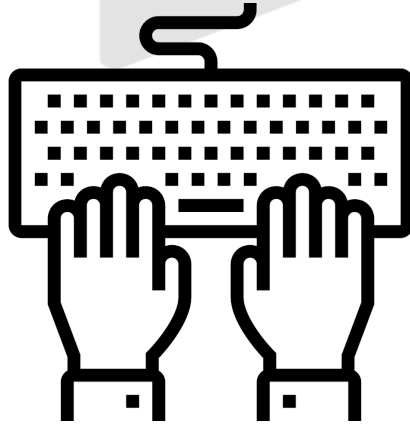
Realities of great software projects...



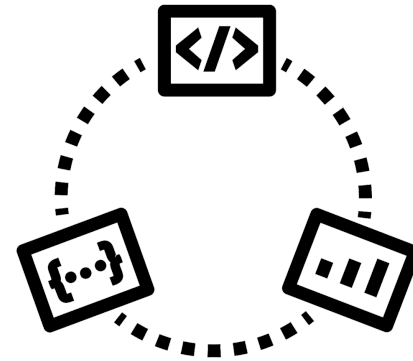
Research



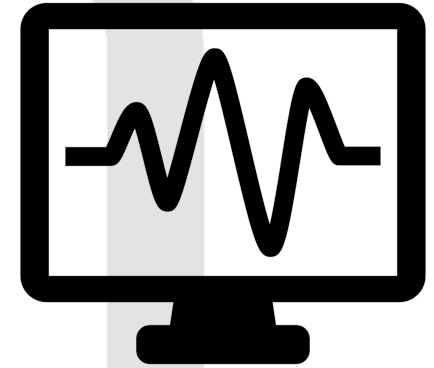
Design



Develop



Deploy



Monitor

My audience: developers



Different developers, different needs



“This is a job” developer



I work for a company that defines our tools and computer setup. I know there are probably better things out there, but as I don't get to use them at work, why bother learning them?

The “switched on” developer



It is frustrating that there are still people out there with old computers and outdated browsers. With a library I can support them as I'd rather learn and try out new things.

The “I am here to deliver” developer



My clients don't pay extra for fancy new technology. They need to get something that works on their machines. It needs to be legally compliant, you know, accessible so they can't be sued. That's why I use a framework.

The “the web is broken” developer

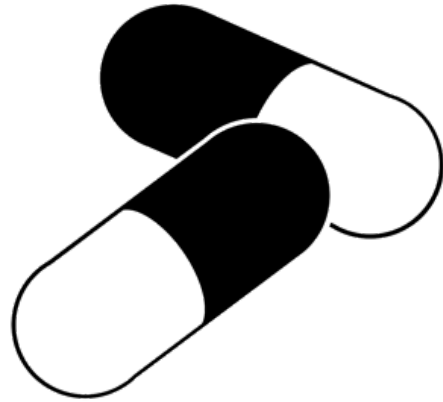


I've always worked with higher programming languages and rich UI environments. The web seems like a mess and I don't want to spend time trying to guess what the outcome of my code is.

The “I’m here to learn” developer



I just started as a web developer and wherever I look, I am confused. There is so much information and often people give contradicting advice. I just want to learn the right thing...



Somehow bitter
pills to swallow...



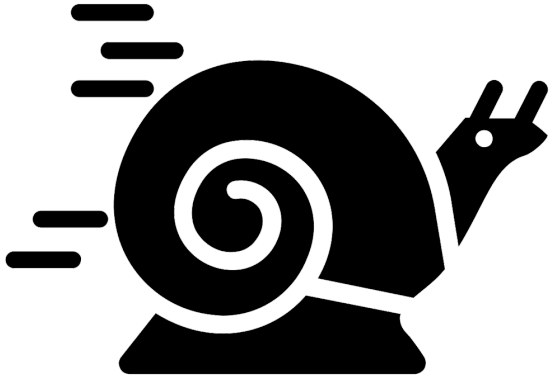
- The caring developer is a tiny fraction of the market
- For many, this is a job to make a living
- New developers don't find an empathic, open world, but an avalanche of choice and opinions
- Satisfying developer needs has no direct correlation to creating better user products



Tech innovation
isn't the answer



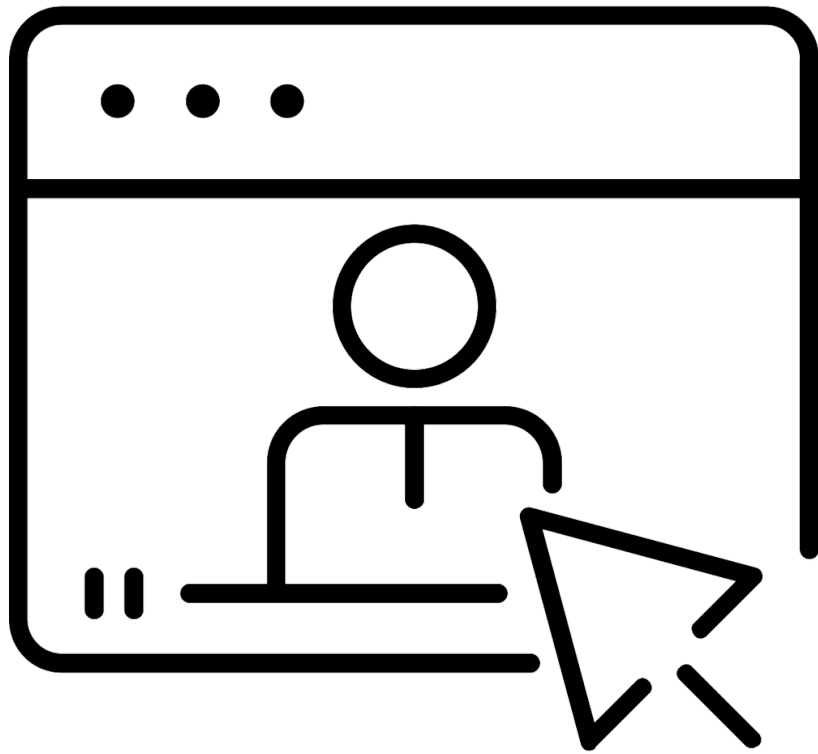
- Constant change in the developer space is a double-edged sword
- We empower a small group of highly motivated people at the cost of overwhelming others
- We don't find out about real problems as we don't ask people outside our bubble



Why are things
"slow"?

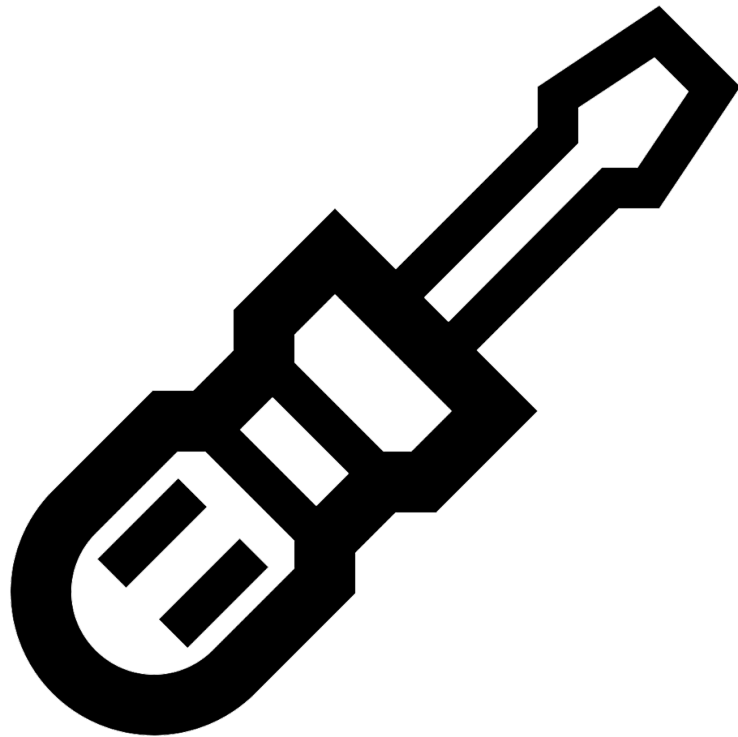
- Security
- Privacy
- Performance
- Maintainability
- Ownership
- Backwards compatibility

Browsers for developers



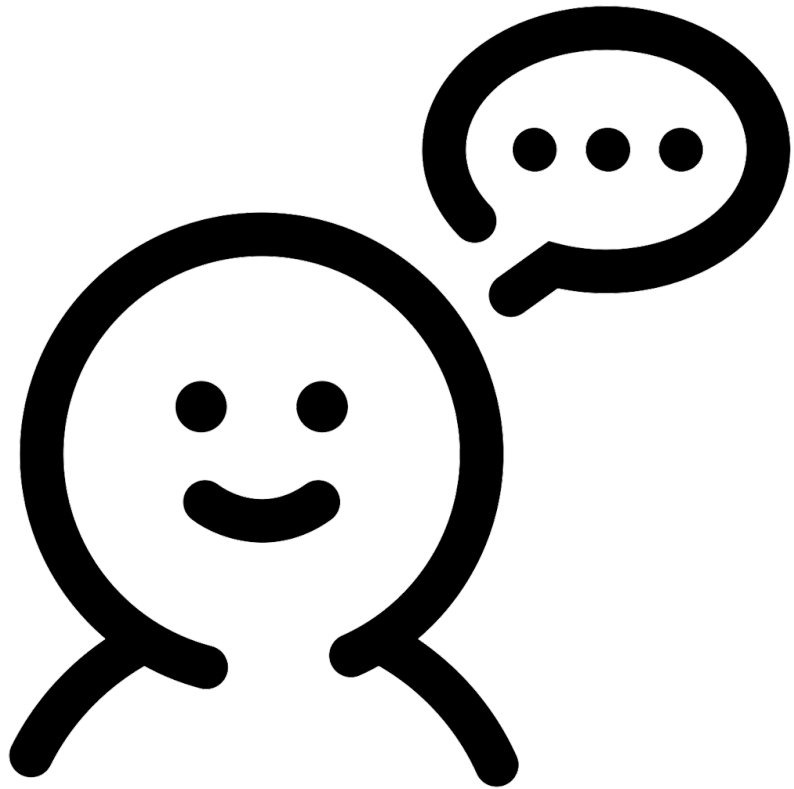
- Consumption and creation platform
- App platform (Progressive Web Apps, Electron...)
- Test platform (Automation with Puppeteer / Playwright)
- Preview of technology of the web to come (Beta, Canary, Developer builds, flags, origin trials)

Why do we need web standards?



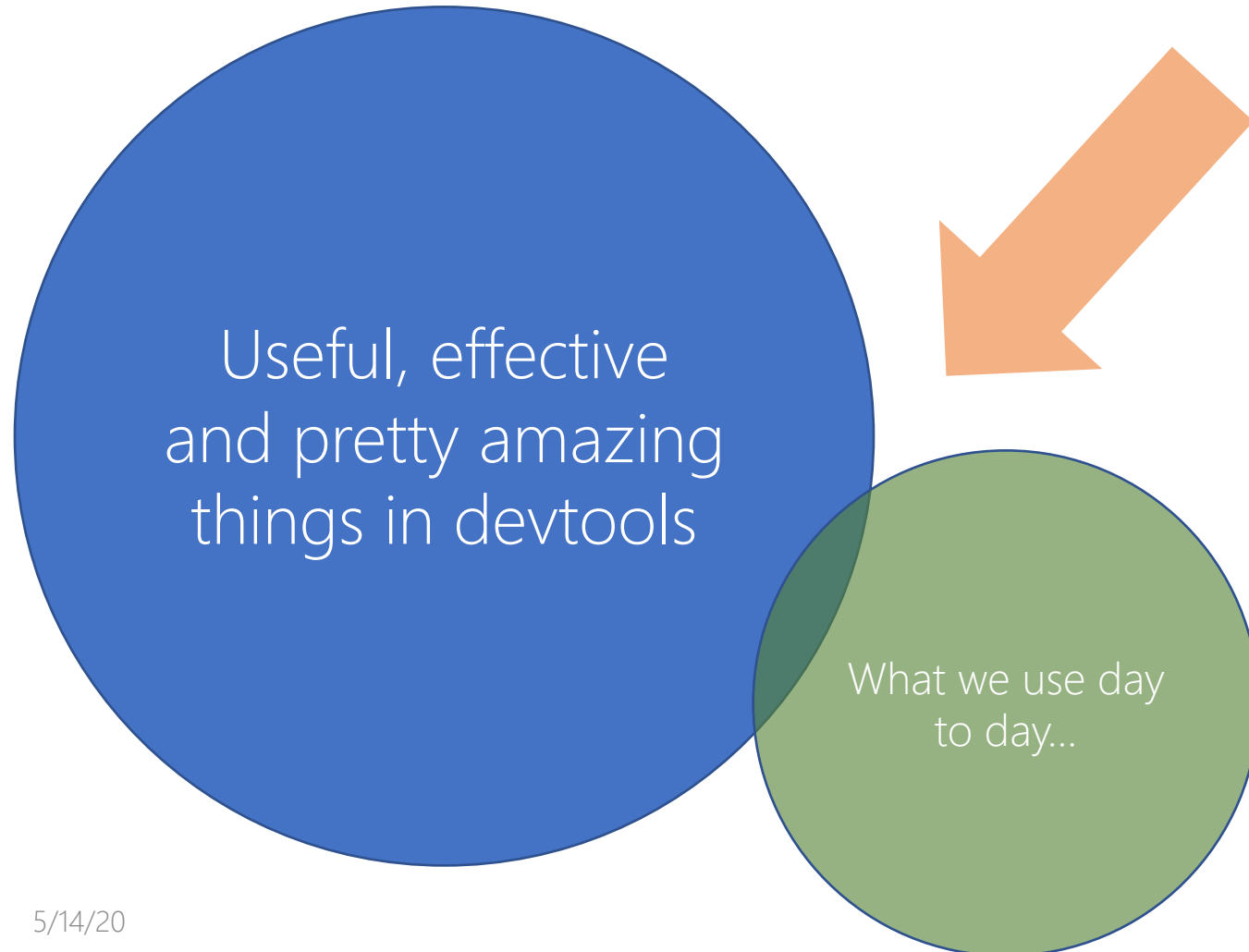
- Predictability of code outcome
- Prevention of monopolies
- Quality control – if your code validates the issue must be somewhere else
- Maintainability, backwards compatibility and future readiness.

Web standards are an agreement

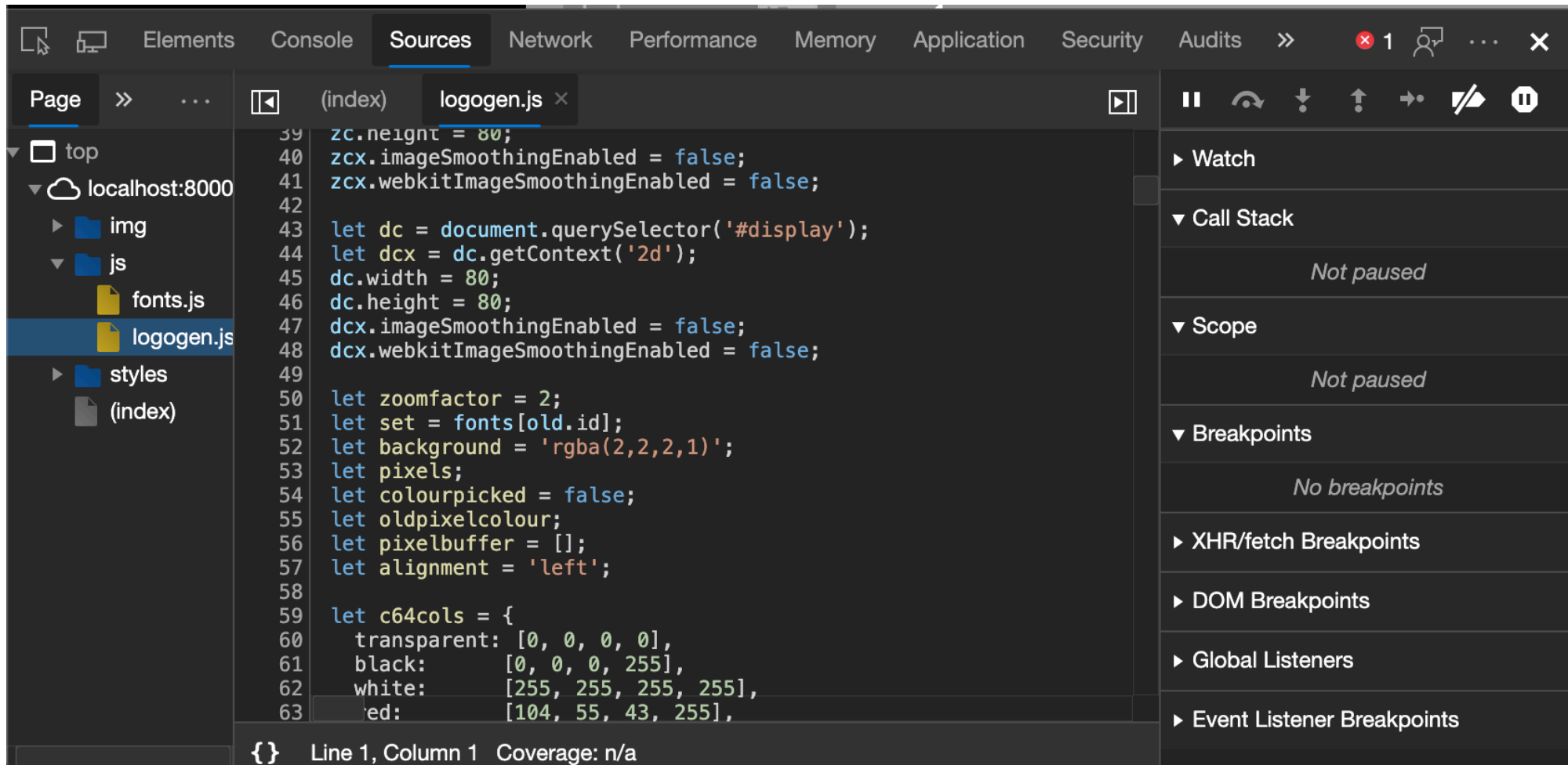


- They used to be pretty academic
- Nowadays they are informed by use and needs of developers and users collected by browser makers and developers
- Discussions happen in the open and you are welcome to join

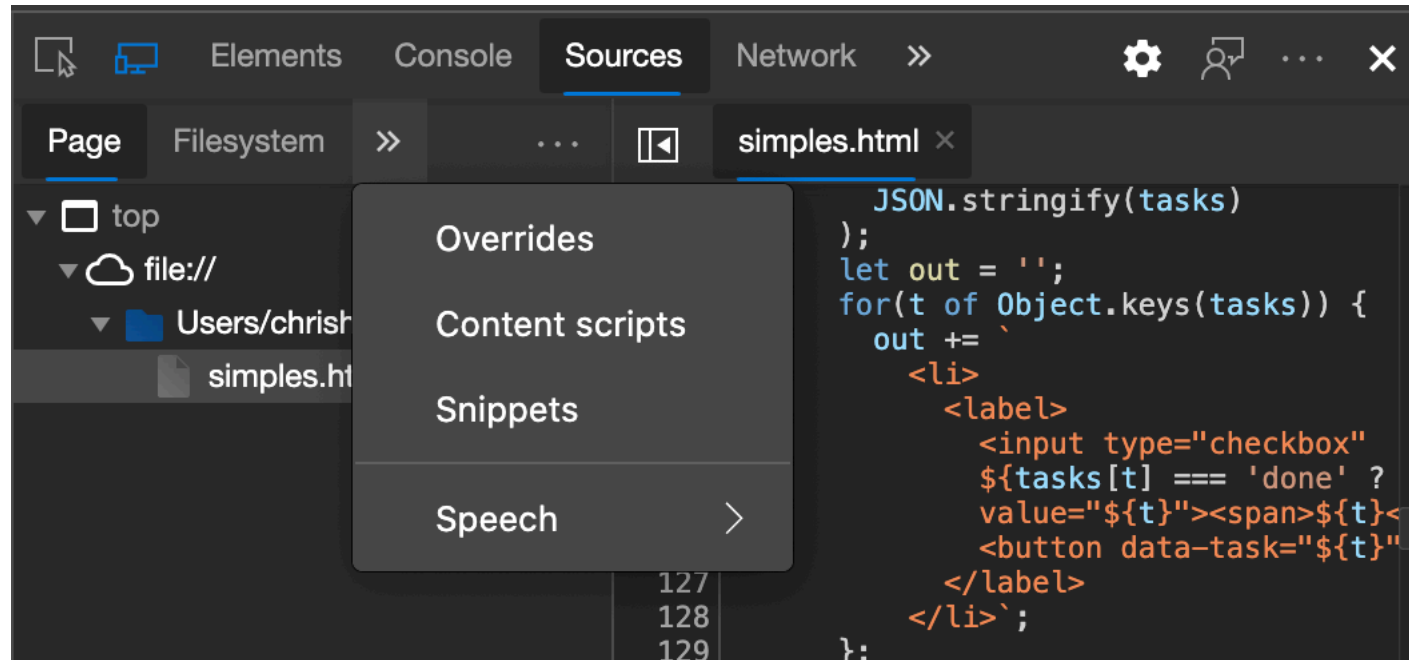
Usage gap



Editor in the browser...



Editor in the browser...



Upgrading your dev experience...





Breakpoints >
console.log()

- You can set breakpoints in the editor in the browser
- You can also set breakpoints in your text editor (f.e. Visual Studio Code)
- They are more work upfront than a console.log() - but worth it.



Breakpoints >
`console.log()`

- Your code execution is paused – errors can't slip through
- You get an end-to-end picture of what is happening
- You are not likely to litter the web with yet another console message that end users should never see



DOM breakpoints!



Task:

What do you need to do?

Add

buy cookies

get cookie consent



DOM breakpoints?



Task:

What do you need to do?

Add

buy cookies

get cookie con

- ← Back ⌘[
- Forward ⌘]
- 🔄 Refresh ⌘R
- 📄 Save As... ⌘S
- 🖨️ Print... ⌘P
- 📺 Cast Media to Device...
- A) Read Aloud ⌘U
- 🗣️ Translate to English
- 📁 Add Page to Collections >
- View Page Source ⌘U
- View Frame Source
- Refresh Frame
- 🗨️ Inspect ⌘I

Responsive ▾

852

x

738

70% ▾

Online ▾



Elements

Console

Sources

Network

Performance

Memory



Task:

What do you need to do?

Add

buy cookies

get cookie consent

li

490.39 x 42

ACCESSIBILITY

Name

Role

listitem

Keyboard-focusable

```
<!DOCTYPE html>
<html lang="en">
  <head>...</head>
  <body>
    <form>
      <div class="searchbar">...</div>
      <ul id="tasks">
        <li>...</li>
        ...
        <li == $0
          <label>
            <input type="checkbox" value="get cookie consent">
            <span>get cookie consent</span>
            <button data-task="get cookie consent">x</button>
          </label>
        </li>
      </ul>
    </form>
  </body>
</html>
```

html body form ul#tasks li label span

Styles

Event Listeners

DOM Breakpoints

Properties

Accessibility

Filter

:hov

.cls

+

```
element.style {
}
```

```
li {
  list-style: none;
  padding: 5px;
  line-height: 1.2;
}
```

[simples.html:50](#)

position 0

margin -

border -

padding 5



Console

3D View

What's New

WebAudio

Rendering

Issues

Changes





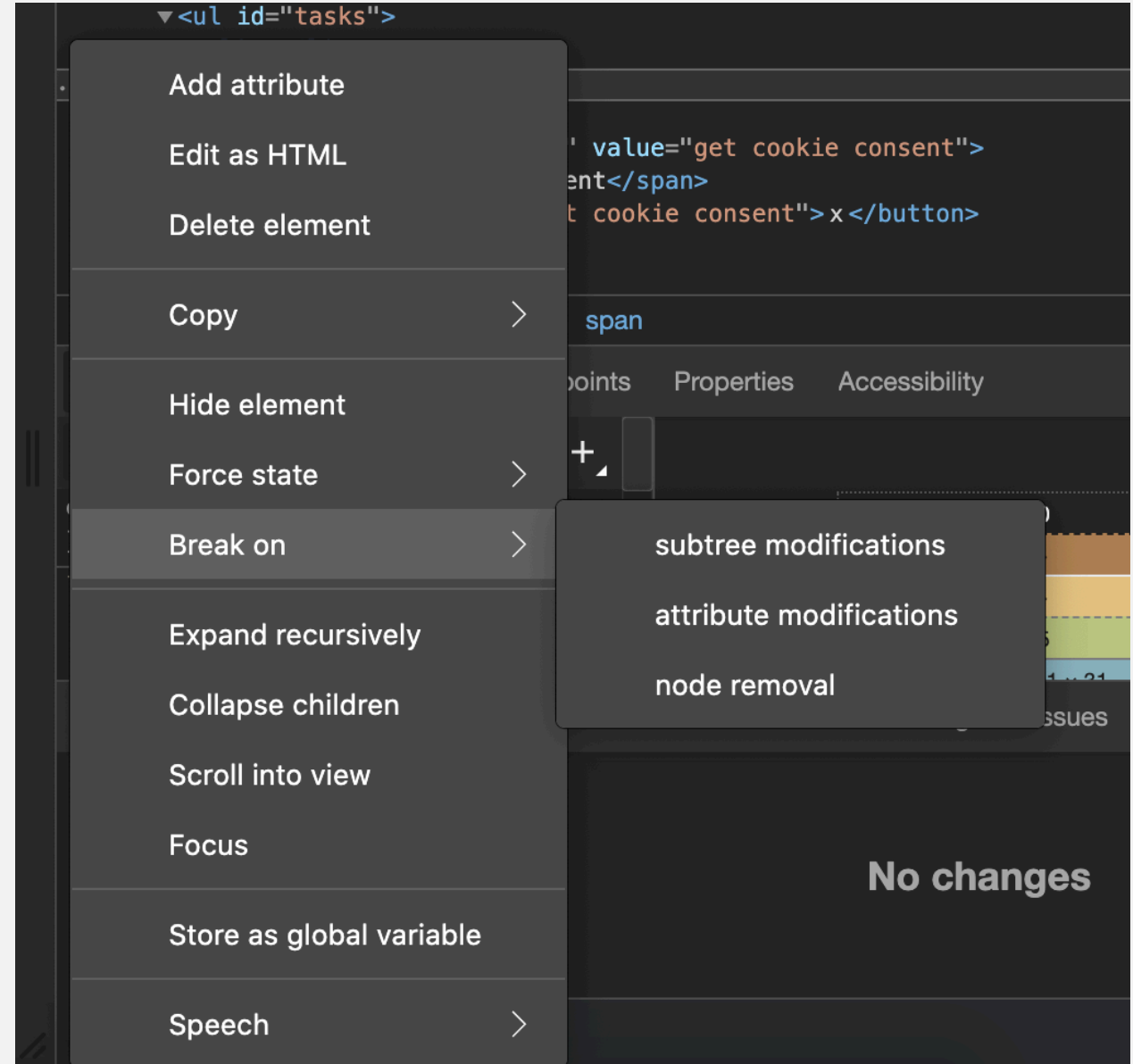
DOM breakpoints!



```
▼ <ul id="tasks">  
  ▶ <li>...</li>  
  ...  
  ▼ <li> == $0  
    ▼ <label>  
      <input type="checkbox" value="get cookie consent">  
      <span>get cookie consent</span>  
      <button data-task="get cookie consent">x</button>  
    </label>  
  </li>
```



DOM breakpoints!



Responsive ▾ 852 x 738 70% ▾ Online ▾

Task: Paused in debugger ▶ ↺ Add

buy cookies ✕

get cookie consent

Page Filesystem >> ... ◀ simples.html x

- top
 - file://
 - Users/chrisheilmann/Downloads
 - simples.html

```
116     JSON.stringify(tasks)
117   );
118   let out = ''; out = "<li><label><input ty
119   for(t of Object.keys(tasks)) {
120     out += ` out = "<li><label><input type=
121     <li>
122       <label>
123         <input type="checkbox"
124         ${tasks[t] === 'done' ? 'checked' : ''}
125         value="${t}"><span>${t}</span>
126         <button data-task="${t}">x</button>
127       </label>
128     </li>`;
129   };
130   list.innerHTML = out;
131 };
132
133 const addTask = e => {
134   if (task.value) {
135     tasks[task.value] = 'active';
136     updateList();
137     task.value = '';
138   }
139 }
```

{ } Line 130, Column 22 Coverage: n/a

Elements Console Sources Network Performance Memory Application Security Lighthouse Layers

Page Filesystem >> ... simples.html x

top
file://
Users/chrisheilmann/Download
simples.html
inspector://

```
142     const changeTask = e => { e = MouseEvent {isTrusted: true, scre
143       let t = e.target; t = button {disabled: false, form: form, fo
144       if (t.dataset.task) {
145         delete tasks[t.dataset.task];
146         updateList();
147         e.preventDefault();
148       }
149       if (t.nodeName.toLowerCase()=== 'input') {
150         tasks[t.value] = t.checked ? 'done' : 'active';
151         updateList();
152         e.preventDefault();
153       }
154     }
155
156
```

Pause on caught exceptions

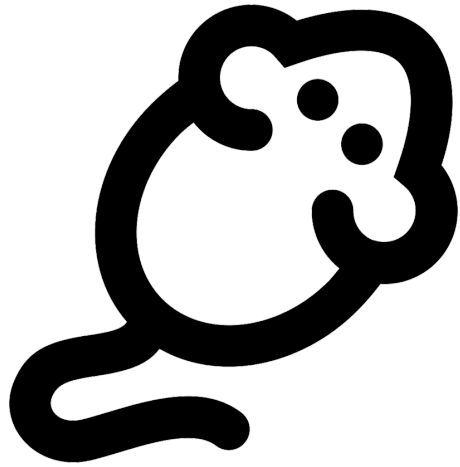
Call Stack

- changeTask simples.html:144

Scope

- Local
 - e: MouseEvent {isTrusted: t...
 - t: button
 - this: undefined
- Script
- Global Window

{ } Line 144, Column 13 Coverage: n/a



No need for a
mouse!



Shortcuts

`⌘ ↓` Decrement by 0.1

Debugger

`F8` `⌘ \` Pause/ Continue

`F10` `⌘ '` Step over

`F11` `⌘ ;` Step into

`⇧ F11` `⌘ ⇧ ;` Step out

`Ctrl .` `Ctrl ,` Next/previous call frame

`Ctrl ⇧ E` Evaluate selection in
console

`Ctrl ⇧ A` Add selection to watch

`⌘ B` Toggle breakpoint

`⌘ ⇧ B` Toggle breakpoint
enabled

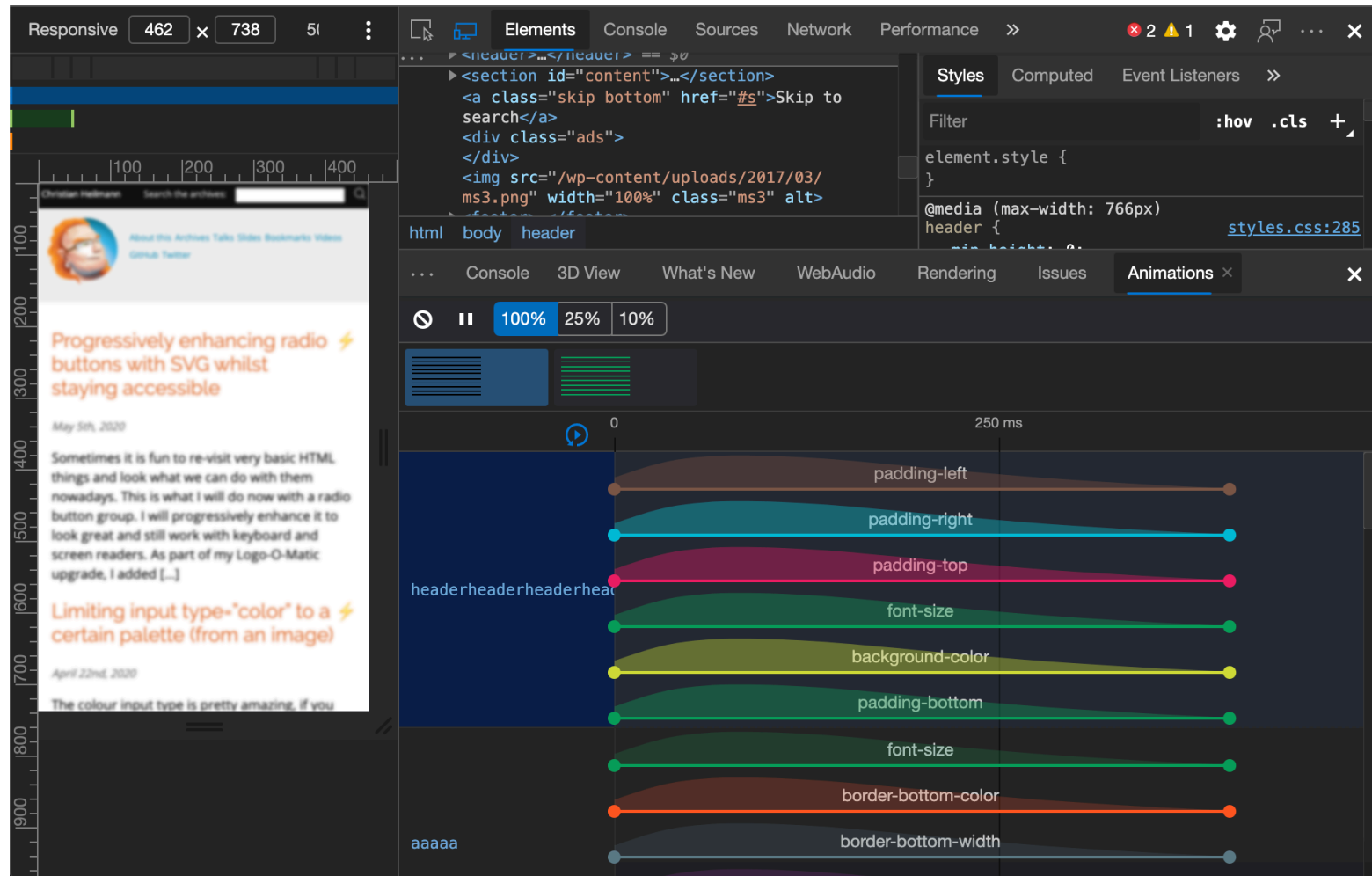
`⌘ F8` Toggle all breakpoints

`⌘ ⌘ B` Open breakpoint editor

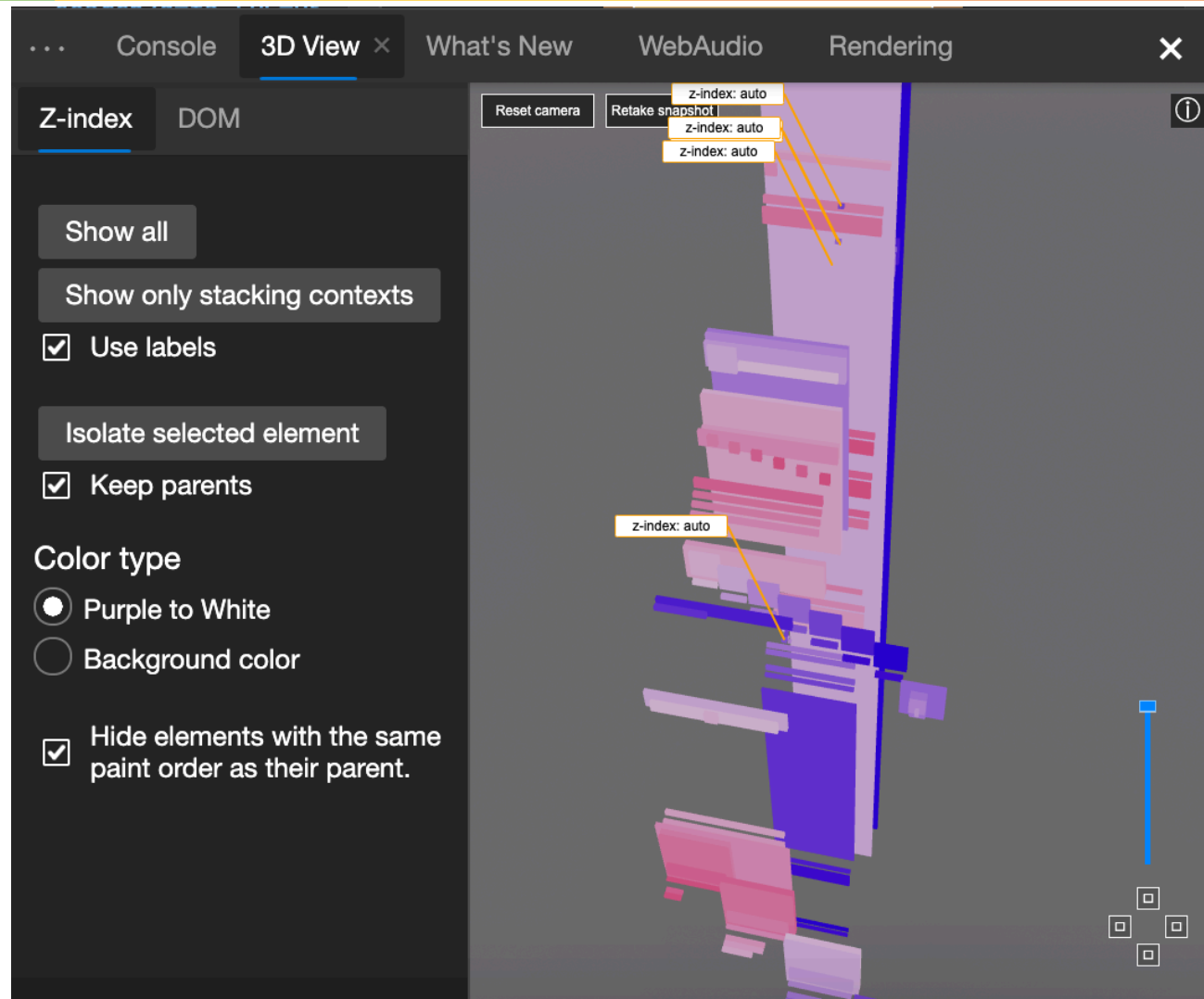
More visual tooling?



Animation Editor








3D View



Easy to miss extra features...

```
.searchbar { simples.html:25  
  display: flex;  
  color: ■ #fff;  
  background: ▶ □ #333;  
  border-radius: ▶ 10px;  
  box-shadow: □ 0 2px 6px ■ #999;  
}
```

```
.searchbar { simples.html:25  
✓ display: flex;  
✓ color: ■ #fff;  
✓ background: ▶ □ #333;  
✓ border-radius: ▶ 10px;  
✓ box-shadow: □ 0 2px 6px ■ #999;  
}     
```

Colour Picker

```
.searchbar { simples.html:25  
  display: flex;  
  color: ■ #fff;  
  background: ▶ ■ #333;  
  border-radius: ▶ 10px;  
  box-shadow: □ 0 2px 6px ■ #999;  
}
```

```
.searchbar { simples.html:25  
  ✓ display: flex;  
  ✓ color: ■ #fff;  
  ✓ background: ▶ ■ #333;  
  ✓ border-radius: ▶ 10px;  
  ✓ box-shadow: □ 0 2px 6px ■ #999;  
}
```

The image shows a browser's developer tools interface. A color picker is open over a searchbar element. The picker displays a color gradient and a hex code input field set to #ffffff. The background shows the CSS styles for the searchbar element, including display: flex, color: #fff, background: #333, border-radius: 10px, and box-shadow: 0 2px 6px #999.

Shadow Editor

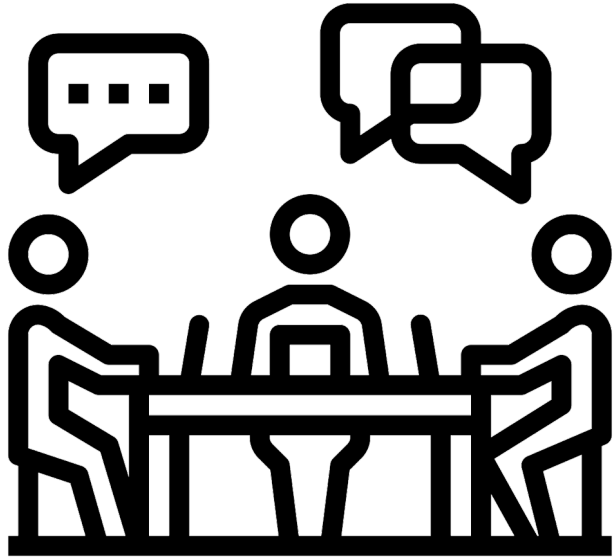
The image shows a browser window with a search bar containing the text "What do you need to do?" and an "Add" button. A shadow editor overlay is positioned over the search bar, displaying the following CSS code:

```
.searchbar {  
  display: flex;  
  color: #fff;  
  background-color: #333;  
  border-radius: 10px;  
  box-shadow: 10px -11px 6px #999;  
  background-color: white;  
}
```

The shadow editor interface includes the following controls:

- Type:** Outset (selected) / Inset
- X offset:** 10px
- Y offset:** -11px
- Blur:** 6px (with a slider)
- Spread:** 0 (with a slider)

The editor also features a visual diagram of the shadow effect, showing a blue dot representing the shadow's origin and a white dot representing the element's position. The background of the editor shows the browser's developer tools with the CSS styles for the search bar element.



Currently in
production:



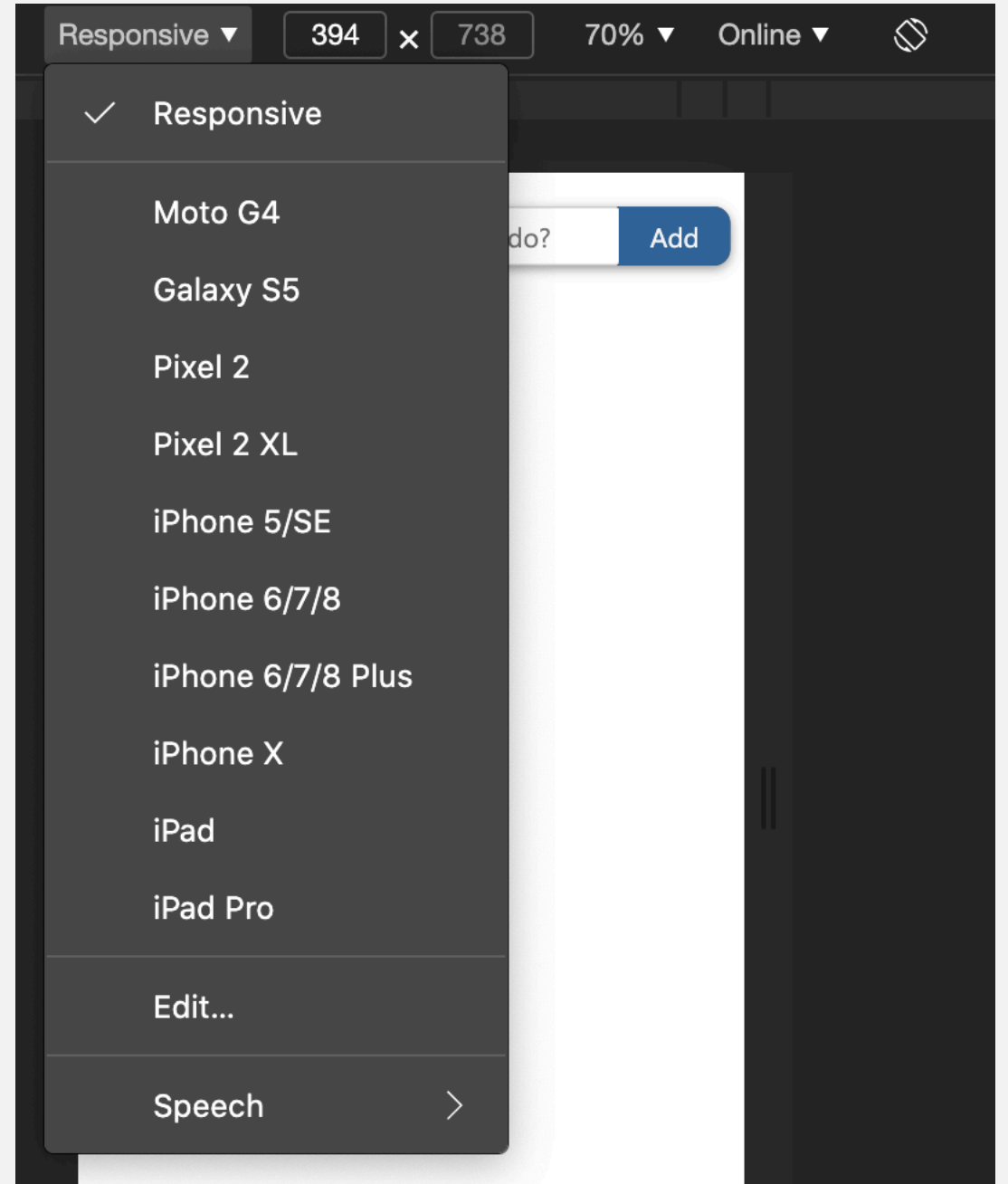
- CSS Grid tooling
- Font tooling
- Service Worker lifecycle visualization
- ??? (you decide)

Emulation



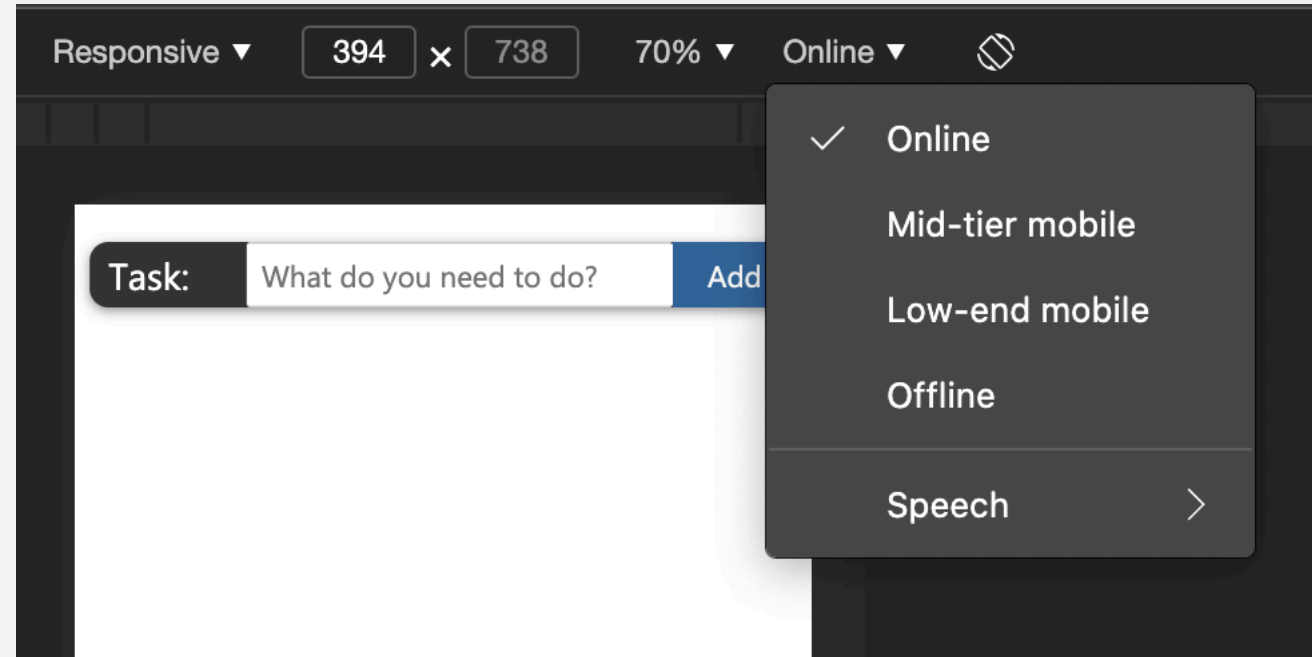


Device Mode



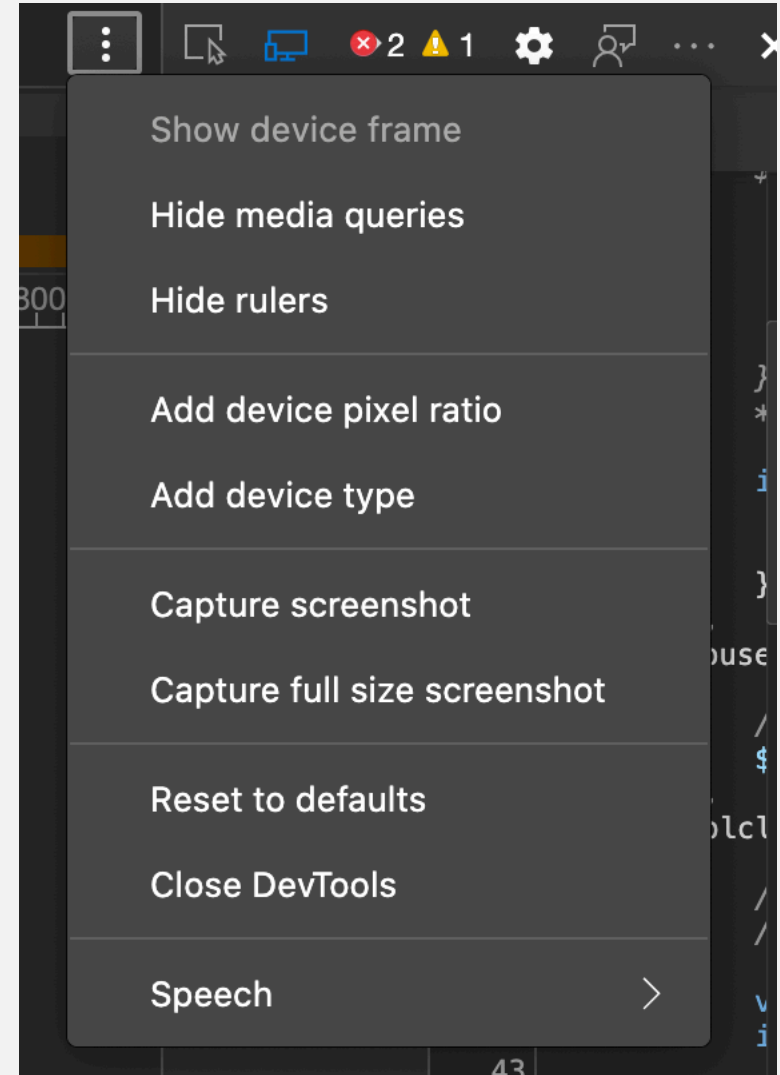


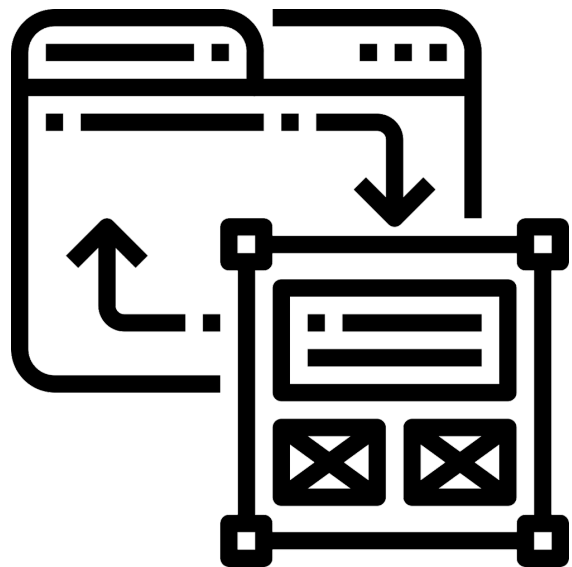
Device Mode



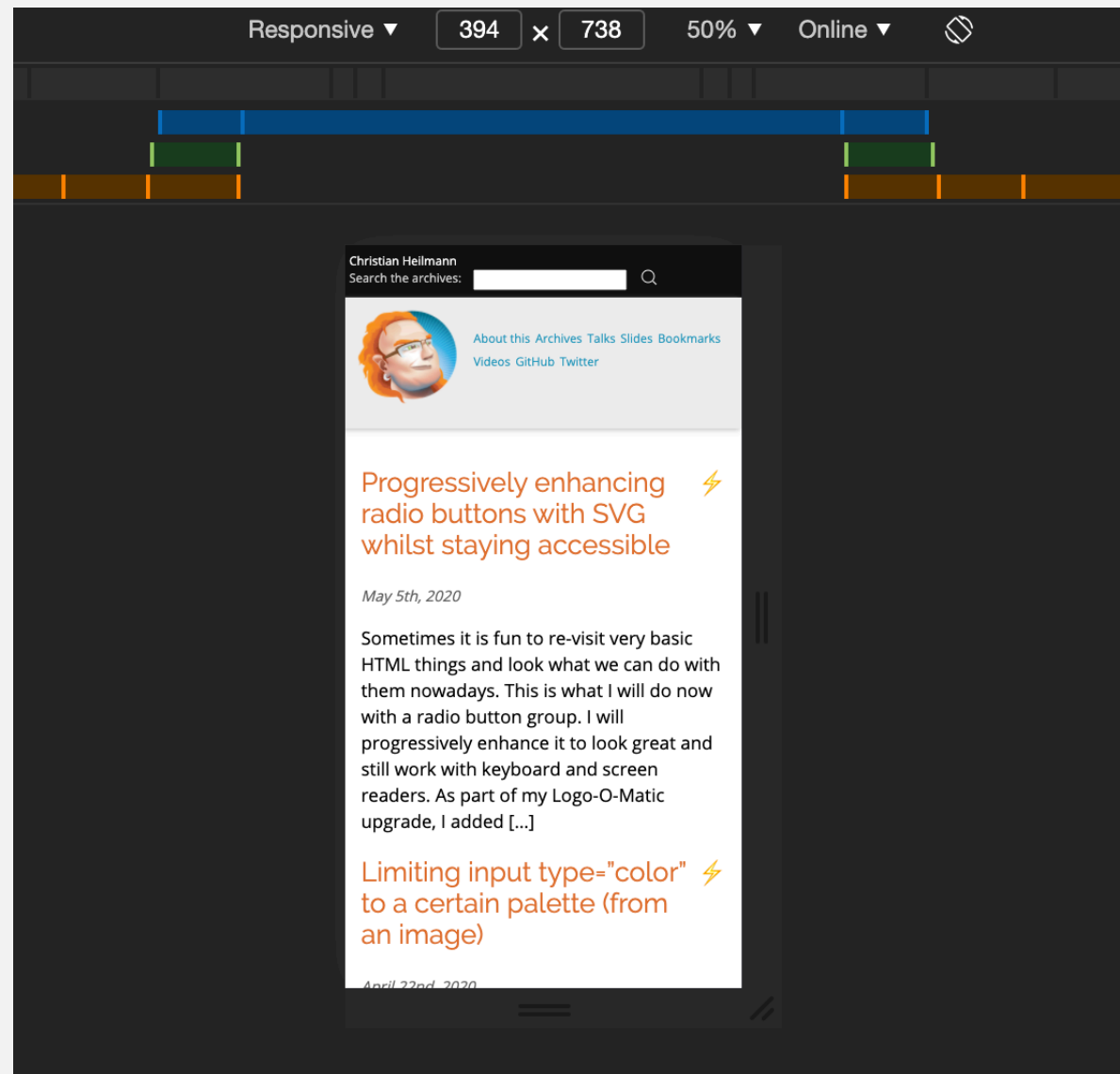


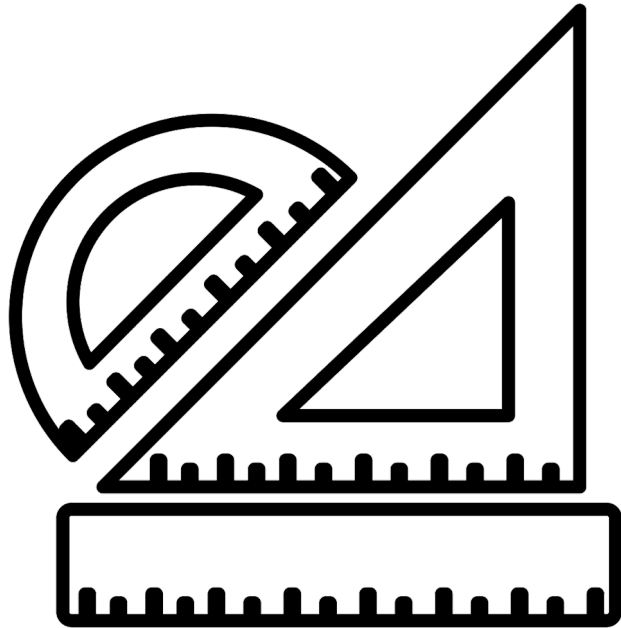
More features of Device Emulation



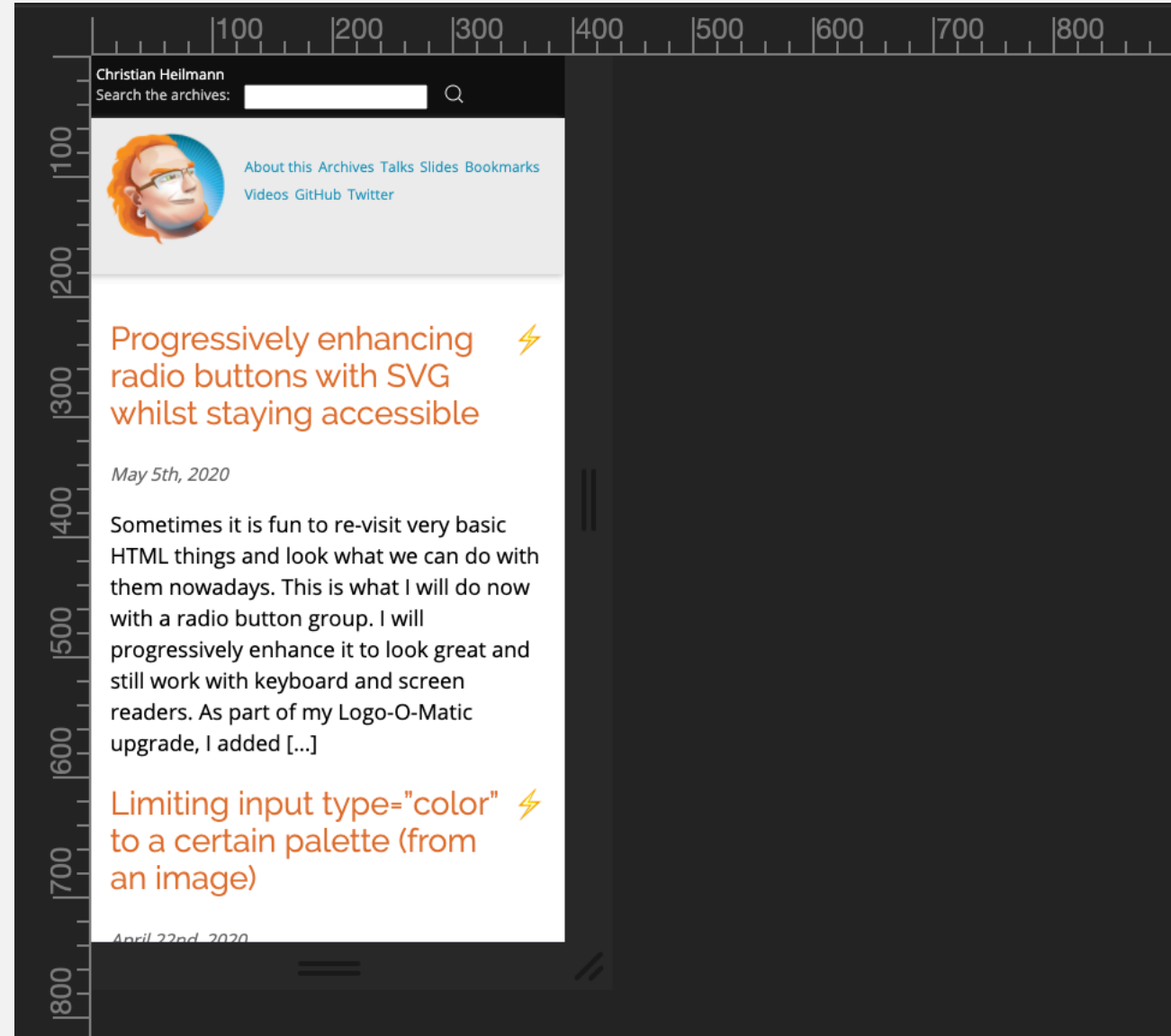


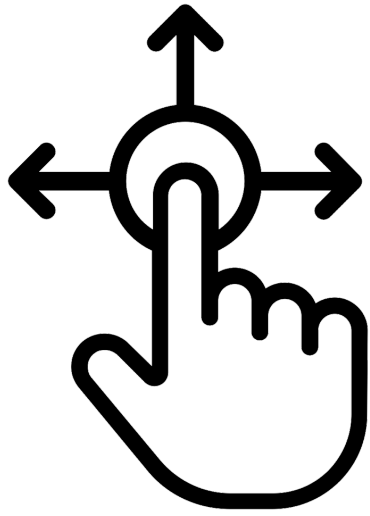
Media Queries





Screen Ruler





Interaction emulation



```
Elements Console Sources Network Performance
<!DOCTYPE html>
<html lang="en">
  <head>...</head>
  <body>
    <form>
      <div class="searchbar">...</div>
      <ul id="tasks">
        <li>...</li> == $0
      </ul>
    </form>
  </body>
</html>

html body form ul#tasks li

Styles Event Listeners DOM Breakpoints Properties Accessibility
Filter :hov .cls +
Force element state
 :active
 :focus
 :focus-within
 :hover
 :visited
element.style {
}
li { inspector-stylesheet:1
}
li { simples.html:50

position
margin
border
pad
```

Task:

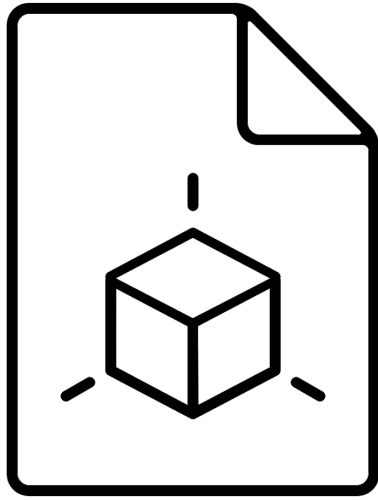
get cookie consent

```
<!DOCTYPE html>
<html lang="en">
  <head>...</head>
  <body>
    <form>
      <div class="searchbar">...</div>
      <ul id="tasks">
```

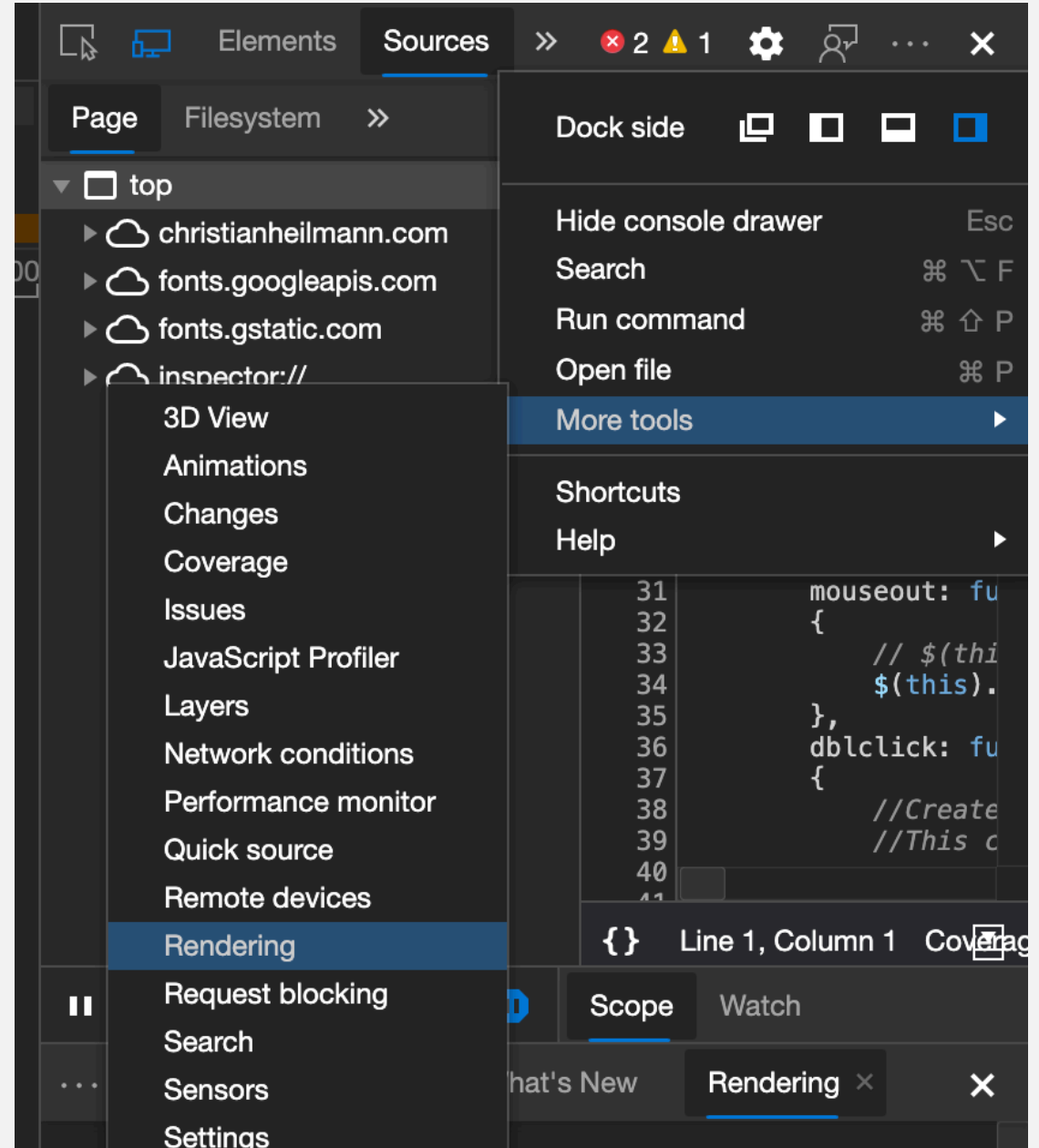
- Add attribute
- Edit as HTML
- Delete element
- Copy >
- Hide element
- Force state >
- Break on >
- Expand recursively
- Collapse children
- Scroll into view
- Focus
- Store as global variable
- Speech >

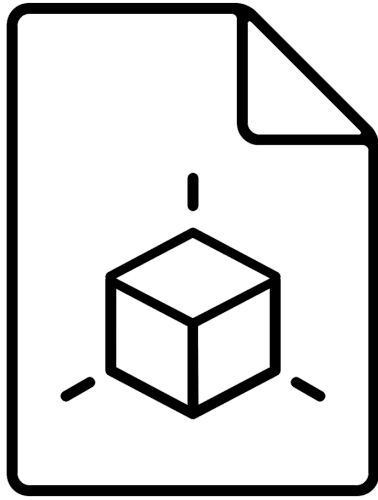
- Properties Accessibility
- :active
- ✓ :hover
- :focus
- :visited
- :focus-within

No chang



Rendering menu





Rendering menu



... Console 3D View What's New **Rendering** ×

- Paint flashing**
Highlights areas of the page (green) that need to be repainted. May not be suitable for people prone to photosensitive epilepsy.
- Layout Shift Regions**
Highlights areas of the page (blue) that were shifted. May not be suitable for people prone to photosensitive epilepsy.
- Layer borders**
Shows layer borders (orange/olive) and tiles (cyan).
- FPS meter**
Plots frames per second, frame rate distribution, and GPU memory.
- Scrolling performance issues**
Highlights elements (teal) that can slow down scrolling, including touch & wheel event handlers and other main-thread scrolling situations.
- Highlight ad frames**
Highlights frames (red) detected to be ads.
- Hit-test borders**
Shows borders around hit-test regions.

Emulate CSS media type
Forces media type for testing print and screen styles

No emulation ▾

Emulate CSS media feature prefers-color-scheme
Forces CSS media feature prefers-color-scheme



Emulate CSS media type
Forces media type for testing print and screen styles

No emulation
print
screen

feature prefers-color-scheme
Forces CSS prefers-color-scheme media feature



... Console 3D View What's New **Rendering** ×

Paint flashing
Highlights areas of the page (green) that need to be repainted. May not be suitable for people prone to photosensitive epilepsy.

Layout Shift Regions
Highlights areas of the page (blue) that were shifted. May not be suitable for people prone to photosensitive epilepsy.

Layer borders
Shows layer borders (orange/olive) and tiles (cyan).

FPS meter
Plots frames per second, frame rate distribution, and GPU memory.

Scrolling performance issues
Highlights elements (teal) that can slow down scrolling, including touch & wheel event handlers and other main-thread scrolling situations.

Highlight ad frames
Highlights frames (red) detected to be ads.

Hit-test borders
Shows borders around hit-test regions.

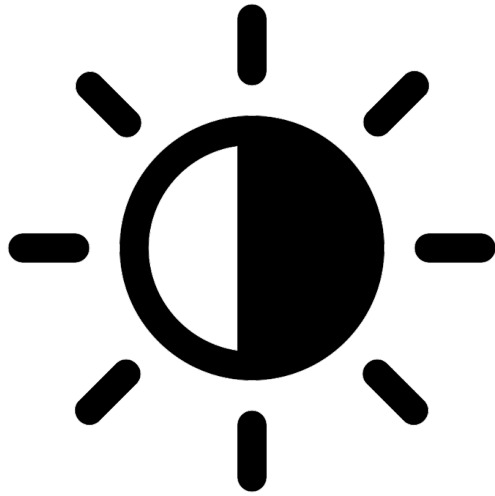
Emulate CSS media type
Forces media type for testing print and screen styles

No emulation ▾

Emulate CSS media feature prefers-color-scheme
Forces CSS prefers-color-scheme media feature

Accessibility concerns





Emulate CSS media feature prefers-color-scheme
Forces CSS prefers-color-scheme media feature

✓ No emulation

prefers-color-scheme: light

prefers-color-scheme: dark

-reduced-motion

Forces CSS prefers-reduced-motion media feature



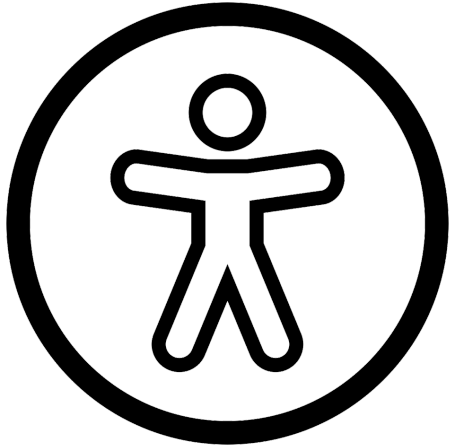
... Console 3D View What's New **Rendering** ×

- Paint flashing**
Highlights areas of the page (green) that need to be repainted. May not be suitable for people prone to photosensitive epilepsy.
- Layout Shift Regions**
Highlights areas of the page (blue) that were shifted. May not be suitable for people prone to photosensitive epilepsy.
- Layer borders**
Shows layer borders (orange/olive) and tiles (cyan).
- FPS meter**
Plots frames per second, frame rate distribution, and GPU memory.
- Scrolling performance issues**
Highlights elements (teal) that can slow down scrolling, including touch & wheel event handlers and other main-thread scrolling situations.
- Highlight ad frames**
Highlights frames (red) detected to be ads.
- Hit-test borders**
Shows borders around hit-test regions.

Emulate CSS media type
Forces media type for testing print and screen styles

No emulation ▾

Emulate CSS media feature prefers-color-scheme
Forces CSS prefers-color-scheme media feature



Emulate CSS media feature prefers-reduced-motion
Forces CSS prefers-reduced-motion media feature

✓ No emulation
prefers-reduced-motion: reduce



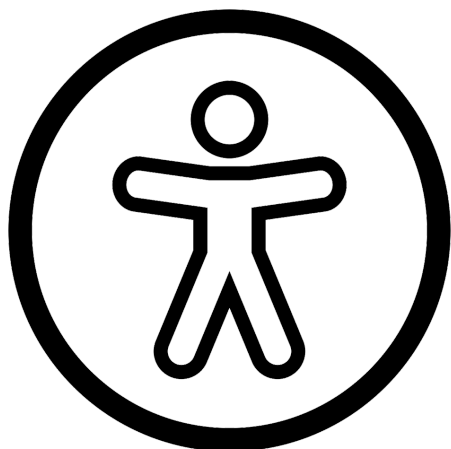
... Console 3D View What's New **Rendering** ×

- Paint flashing**
Highlights areas of the page (green) that need to be repainted. May not be suitable for people prone to photosensitive epilepsy.
- Layout Shift Regions**
Highlights areas of the page (blue) that were shifted. May not be suitable for people prone to photosensitive epilepsy.
- Layer borders**
Shows layer borders (orange/olive) and tiles (cyan).
- FPS meter**
Plots frames per second, frame rate distribution, and GPU memory.
- Scrolling performance issues**
Highlights elements (teal) that can slow down scrolling, including touch & wheel event handlers and other main-thread scrolling situations.
- Highlight ad frames**
Highlights frames (red) detected to be ads.
- Hit-test borders**
Shows borders around hit-test regions.

Emulate CSS media type
Forces media type for testing print and screen styles

No emulation ▾

Emulate CSS media feature prefers-color-scheme
Forces CSS prefers-color-scheme media feature



Emulate vision deficiencies

Forces vision deficiency emulation

Blurred vision



Responsive 394 x 738 51

Elements Console Sources Network

Page Filesystem >> << >>

christianheilmann.com

Line 1, Col 1

scrolling performance issues

- Highlights elements (teal) that can slow down scrolling thread scrolling situations.
- Highlight ad frames Highlights frames (red) detected to be ads.
- Hit-test borders Shows borders around hit-test regions.

Emulate CSS media type Forces media type for testing print and screen styles

No emulation

Emulate CSS media feature prefers-color-scheme Forces CSS prefers-color-scheme media feature

No emulation

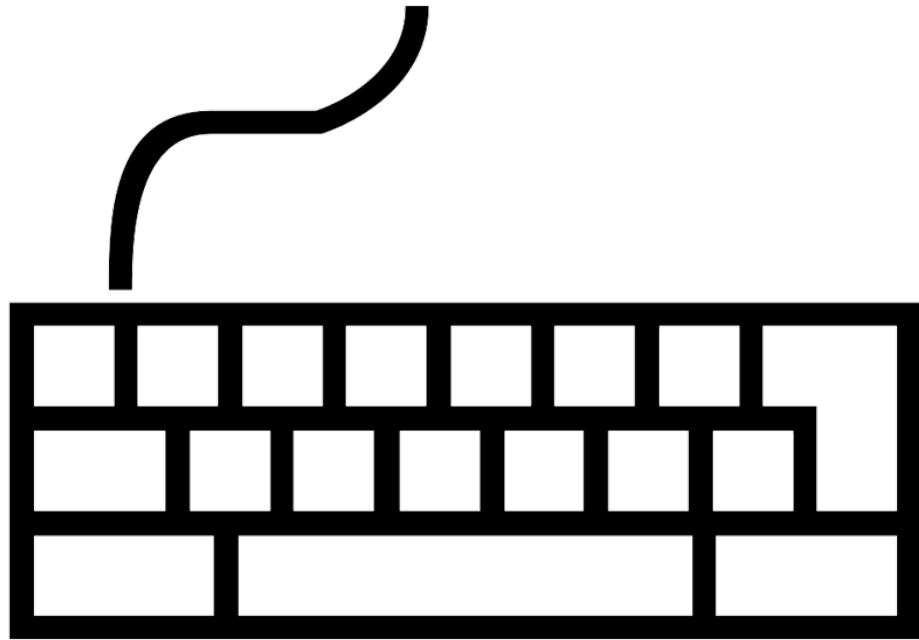
Emulate CSS media feature prefers-reduced-motion Forces CSS prefers-reduced-motion media feature

No emulation

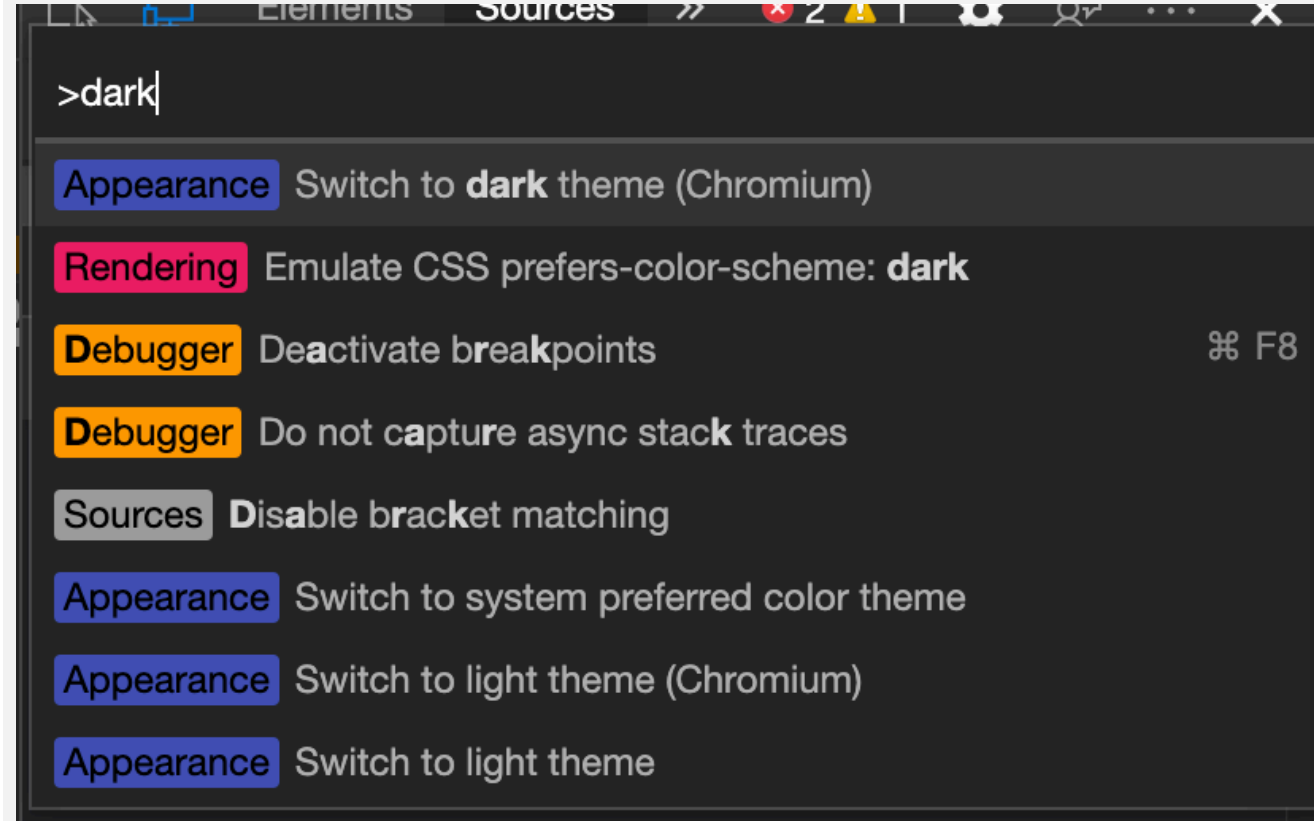
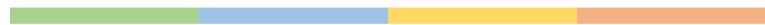
- Blurred vision
- Protanopia
- Deuteranopia
- Tritanopia
- Achromatopsia

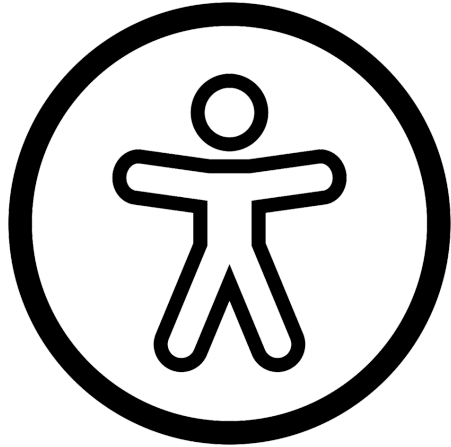
Blurred vision

ncies
cy emulation



Pro tip:
CMD+SHIFT+P

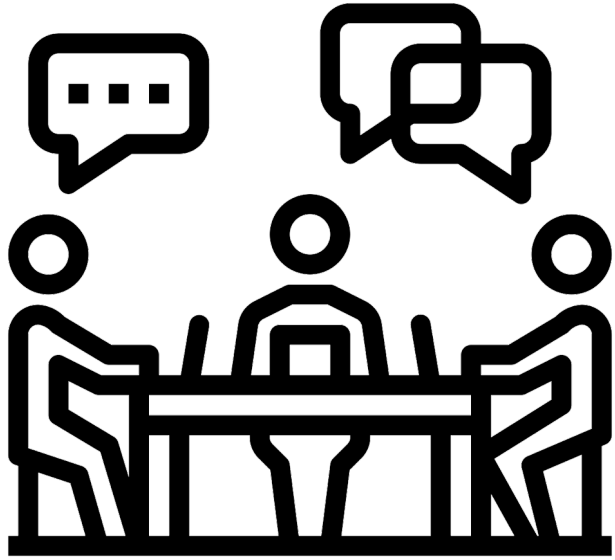




Already shipped:



- Developer tools are keyboard and screenreader accessible
- Developer tools work in high contrast mode



Currently in
production:

- High Contrast simulation
- Simpler accessible interfaces
- Accessibility error reporting in context
- Dual Screen Emulation

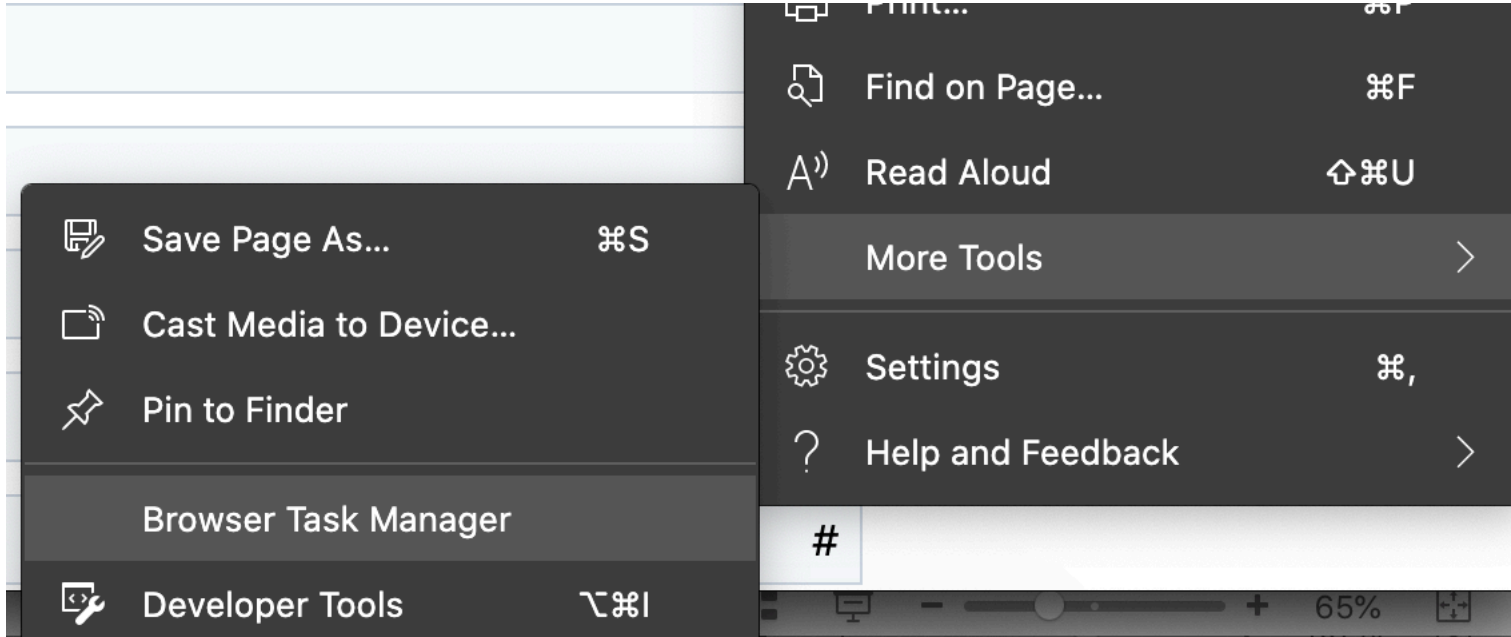
Need for speed?



Process Name	% CPU	CPU Time	Threads	Idle Wake-Ups	% GPU	GPU Time	PID	User
Activity Monitor	12,1	5:45:44,90	6	4	0,0	0,00	416	chrisheilmann
Microsoft PowerPoint	3,7	32:24,81	48	68	0,4	4:16,85	43688	chrisheilmann
Microsoft Edge	3,5	45:51,27	85	5	0,0	0,00	43787	chrisheilmann
Microsoft Teams Helper	3,3	11:54,70	14	179	0,0	0,00	66127	chrisheilmann
Firefox	3,2	6:30:21,10	81	27	0,0	42:09,12	900	chrisheilmann
Microsoft Teams	3,2	13:26,26	40	188	0,0	0,00	66079	chrisheilmann
FirefoxCP Web Content	3,1	3:56:41,37	37	97	0,0	0,00	1106	chrisheilmann
FirefoxCP Web Content	2,2	2:52:56,53	37	24	0,0	0,00	1103	chrisheilmann
iconservicesagent	2,1	1:29,30	5	0	0,0	0,41	370	chrisheilmann
Microsoft Edge	1,8	44:27,67	80	35	0,0	0,00	44060	chrisheilmann
Code Helper (GPU)	1,7	30:15,42	8	12	0,0	6:01,97	56760	chrisheilmann
Electron Helper (Renderer)	1,5	20:33,80	22	15	0,0	0,00	56763	chrisheilmann
Microsoft Outlook	1,1	54:55,94	28	12	0,0	3,90	44503	chrisheilmann
FirefoxCP Web Content	0,8	5:41,07	28	21	0,0	0,00	60390	chrisheilmann
FirefoxCP Web Content	0,7	11:32,43	35	16	0,0	0,00	52567	chrisheilmann
Hemingway Editor	0,6	7:58,25	30	0	0,0	0,00	410	chrisheilmann
DashlaneAgent	0,5	13:20,85	9	17	0,0	0,00	42804	chrisheilmann
Microsoft OneNote	0,5	28:05,53	66	13	0,0	0,00	5461	chrisheilmann
DashlanePluginService	0,4	11:53,77	9	17	0,0	0,00	42805	chrisheilmann
Microsoft Edge Helper	0,4	16:44,42	11	4	0,0	0,00	44069	chrisheilmann
Dropbox	0,4	21:06,45	141	13	0,0	0,00	13070	chrisheilmann
Microsoft Edge Helper (Renderer)	0,4	25:51,26	20	2	0,0	0,00	44938	chrisheilmann
AXVisualSupportAgent	0,3	13:53,72	5	10	0,0	0,00	340	chrisheilmann
FirefoxCP Web Content	0,2	6:16,82	27	4	0,0	0,00	62134	chrisheilmann
sharedfilelistd	0,2	9,59	3	0	0,0	0,00	351	chrisheilmann
Microsoft Edge Helper (Renderer)	0,2	6:00,94	17	10	0,0	0,00	44073	chrisheilmann
FirefoxCP WebExtensions	0,2	23:53,72	28	2	0,0	0,00	914	chrisheilmann
Code	0,2	7:51,25	29	2	0,0	0,00	56759	chrisheilmann

System:	11,94 %		Threads:	4.325
User:	15,06 %		Processes:	600
Idle:	73,00 %			

Browser task manager!



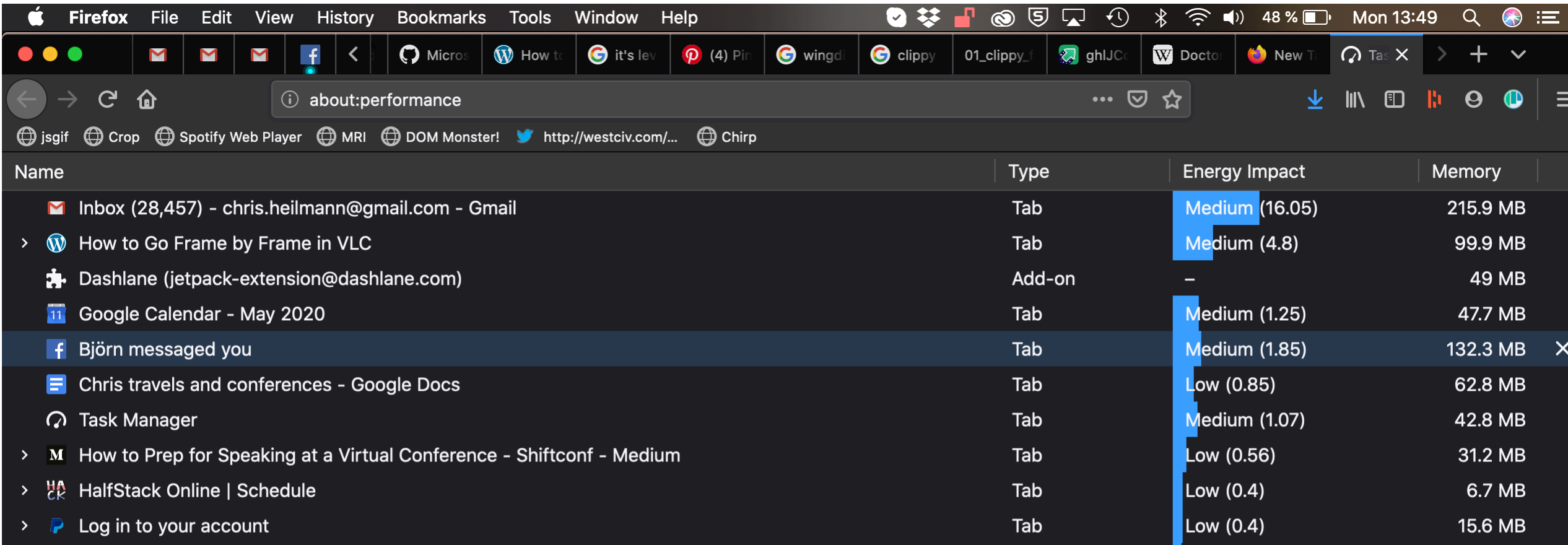
Where did my resources go?

The image shows a browser's context menu on the left and the Browser Task Manager window on the right. The context menu includes options like 'Save Page As...', 'Cast Media to Device...', 'Pin to Finder', 'Browser Task Manager', and 'Developer Tools'. The Browser Task Manager window displays a table of tasks with their respective memory, CPU, and network usage.

Task	Memory	CPU	Network	Process...
Browser	378 MB	5.0	316 B/s	43787
GPU Process	422 MB	0.0	0	43793
Utility: Network Ser	48.9 MB	0.8	0	43794
Utility: Audio Servic	14.3 MB	0.0	0	45203
Tab: DevTools - file	69.5 MB	0.0	0	59291
Extension: Accessil	57.9 MB	0.0	0	43800
Subframe: Accessil				
Subframe: Accessil				
Subframe: Accessil				
Subframe: Accessil				

End Process

Hat tip: also in Firefox!



The screenshot shows the Firefox Performance page with a table of active tabs. The table has four columns: Name, Type, Energy Impact, and Memory. The 'Energy Impact' column is highlighted in blue. The tabs listed include an email inbox, a VLC video, a Dashlane extension, a Google Calendar, a Facebook message, a Google Docs document, a Task Manager window, and several other web pages.

Name	Type	Energy Impact	Memory
Inbox (28,457) - chris.heilmann@gmail.com - Gmail	Tab	Medium (16.05)	215.9 MB
> How to Go Frame by Frame in VLC	Tab	Medium (4.8)	99.9 MB
Dashlane (jetpack-extension@dashlane.com)	Add-on	-	49 MB
Google Calendar - May 2020	Tab	Medium (1.25)	47.7 MB
Björn messaged you	Tab	Medium (1.85)	132.3 MB
Chris travels and conferences - Google Docs	Tab	Low (0.85)	62.8 MB
Task Manager	Tab	Medium (1.07)	42.8 MB
> How to Prep for Speaking at a Virtual Conference - Shiftconf - Medium	Tab	Low (0.56)	31.2 MB
> HalfStack Online Schedule	Tab	Low (0.4)	6.7 MB
> Log in to your account	Tab	Low (0.4)	15.6 MB

How the sausage is made...



Innovation in the open...



Microsoft Edge Explainers

<https://github.com/MicrosoftEdge/MSEdgeExplainers>

Welcome! This repo is home to "explainers" and related documents originating from the Microsoft Edge team.

Jump to section: [Active Explainers](#) | [Alumni](#) | [DevTools](#) | [Other documents](#) | [Withdrawn](#)

Introduction

[Explainers](#) are documents focused on describing a user/ developer/ customer problem (at a high level) and exploring potential solutions. These documents are starting points for engaging in discussion with you and other members of the community. Explainers should address their stated problems in clear and easy to understand language. Proposed solutions should be easy to follow and not too deep in technical details. When you read an explainer, we hope the stated problem is compelling and you can form an opinion for whether the proposed solution would address the problem.

Please provide feedback

We are looking for feedback! Are the stated problems relevant to you? How have they impacted your experience? Do the proposed solutions seem reasonable? Do they follow good [web principles](#)? Would they solve a problem you currently have? (We love to hear that; tell us more about your scenario!) Do you have related use-cases we hadn't considered?

Detailed developer tools explainers

DevTools

We love our developer tools! Checkout these cool innovations being designed for the developer tools:

- [3D View](#)
- [CSS Grid Tooling](#)
- [Customizable Keyboard Shortcuts](#)
- [Dual-screen Emulation](#)
- [High Contrast Simulation](#)
- [Infobar UI Refresh](#)
- [Localization](#)
- [Redux for State](#)
- [Service Worker Improvements](#)
- [Settings Discoverability and Telemetry](#)
- [Settings Search](#)
- [Stackable Overlays](#)
- [webhint](#)

<https://github.com/MicrosoftEdge/MSEdgeExplainers>

Discussion amongst browser makers

The screenshot shows a browser window with the title "3D View for DevTools". The address bar is empty. The page content includes a section titled "Core user stories" with three bullet points:

- A developer can debug z-index by isolating an element from the webpage and following its stacking context hierarchy.
- A developer can find areas of the document that have deep DOM nesting.
- A developer can select a DOM element faster by clicking a box in the 3D View than by expanding the Elements xml tree.

Below the list, the text reads: "The Layers panel has a different purpose than the proposed z-index view. It shows composition layers and its useful to debug repaints, in the screenshot below we can see the same webpage used on the example 1 for the z-index view in page 2. This shows that all the elements were placed inside the same composition layer even when all of them have a different stacking context."

A small screenshot of the Layers panel in DevTools is shown at the bottom left of the page content, displaying a stack of composition layers.

The right-hand sidebar contains a discussion thread:

- Yang Guo** (5 Nov 2019): "What overlap does this feature have with the layers view? Could these two features be merged?"
- Benedikt Meurer** (10 Nov 2019): "I'm also still very much concerned that this will be yet another different thing to worry about for developers. Can we..."

Others join, too!

WebKit / explainers Watch 54 Star 148 Fork 7

[Code](#) [Issues 14](#) [Pull requests 0](#) [Actions](#) [Security 0](#) [Insights](#)

Explainers from WebKit contributors

21 commits | 1 branch | 0 packages | 0 releases | 4 contributors

Branch: master [New pull request](#) [Create new file](#) [Upload files](#) [Find file](#) [Clone or download](#)

hober Separate section for proposals which ahve graduated. Latest commit 90b55cd on 3 Apr

.github/ISSUE_TEMPLATE	Remove issue template for the SMS explainer, which moved to WICG.	last month
IsLoggedIn	Add the IsLoggedIn explainer. (#31)	2 months ago
sms-one-time-code-format	sms-one-time-code-format has moved to WICG.	last month
texttracks	Add Makefiles for easier explainer updating.	3 months ago
.gitignore	Add .gitignore	4 months ago
README.md	Separate section for proposals which ahve graduated.	last month

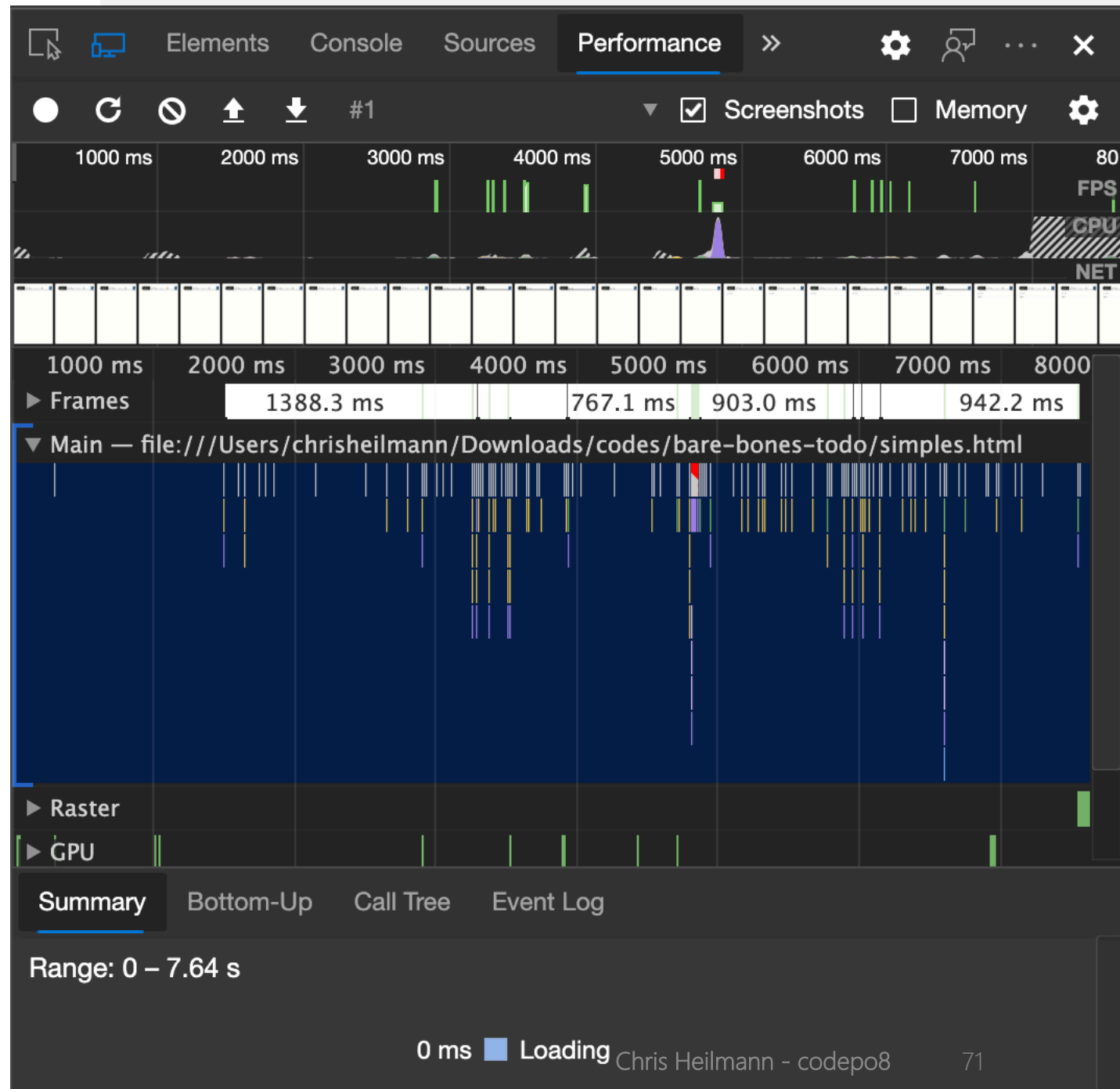
<https://github.com/WebKit/explainers>

Popping the hood...





What are developer tools made of?



W3C Working Draft

TABLE OF CONTENTS

- 1 Introduction
- 2 Foreground Color: the 'color' property
- 3 Representing Colors: the <color> type
 - 3.1 Accessibility and Conveying Information By Color
 - 3.2 Color Spaces of Untagged Colors
 - 3.3 Resolving <color> Values
- 4 sRGB Colors
 - 4.1 The RGB functions: 'rgb()' and 'rgba()'
 - 4.2 The RGB hexadecimal notation: '#RRGGBB'
- 5 Color Keywords
 - 5.1 Named Colors
 - 5.2 System Colors
 - 5.3 The 'transparent' keyword
 - 5.4 The 'currentcolor' keyword
- 6 HSL Colors: 'hsl()' and 'hsla()' functions
 - 6.1 Converting HSL colors to sRGB colors

Authors *may* also use these keywords at any time, but *should* be careful to use the colors in [matching background-foreground pairs](#) to ensure appropriate contrast, as any particular contrast relationship across non-matching pairs (e.g. 'Canvas' and 'ButtonText') is not guaranteed.

The <system-color> keywords are defined as follows:

'Canvas'
Background of application content or documents.

'CanvasText'
Text in application content or documents.

'LinkText'
Text in non-active, non-visited links.

'VisitedText'

Open Developer Tools
(CMD+SHIFT+I on Mac or
F12/CTRL+SHIFT+I on Windows)

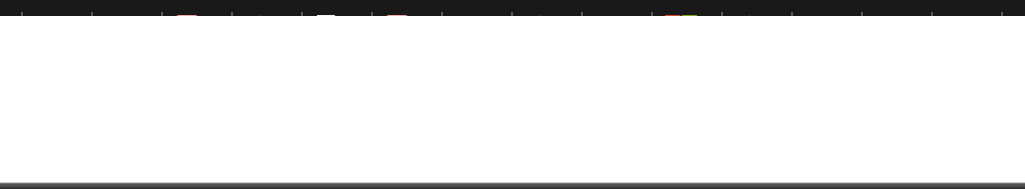
'Field'
Background of input fields.

'FieldText'
Text in input fields.

Working Draft

TABLE OF CONTENTS

- 1 Introduction
- 2 Foreground Color: the 'color' property



Context menu options:

- Dock side... (with a red circle around the icon)
- Show console... (Esc)
- Search (⌘ ⇧ F)
- Run command (⌘ ⇧ P)
- Open file (⌘ P)
- More tools

Elements Console Sources Network Performance

```

<!DOCTYPE html>
<html lang="en">
  <head>...</head>
  <body class="h-entry toc-sidebar"> == $0
    <p id="toc-nav">...</p>
    <div class="head">...</div>
    <div class="p-summary" data-fill-with="abstract">...</div>
    <h2 class="no-num no-toc no-ref heading settled" id="...>...</h2>
    <div data-fill-with="status">...</div>
    <div data-fill-with="at-risk">...</div>
    <nav data-fill-with="table-of-contents" id="toc">...</nav>
    <main>...</main>
    <h2 class="no-ref no-num heading settled" id="...>...</h2>
    <h3 class="heading settled" id="...>...</h3>
    <p>...</p>
    <p>...</p>
    <p>...</p>
    <div class="example" id="example">
      <p>...</p>
      <p class="note" role="note">...</p>
    </div>
    <h3 class="heading settled" id="conform-classes">...</h3>
    <p>Conformance to this specification is defined for three conformance classes: </p>
    <dl>...</dl>
  </body>

```

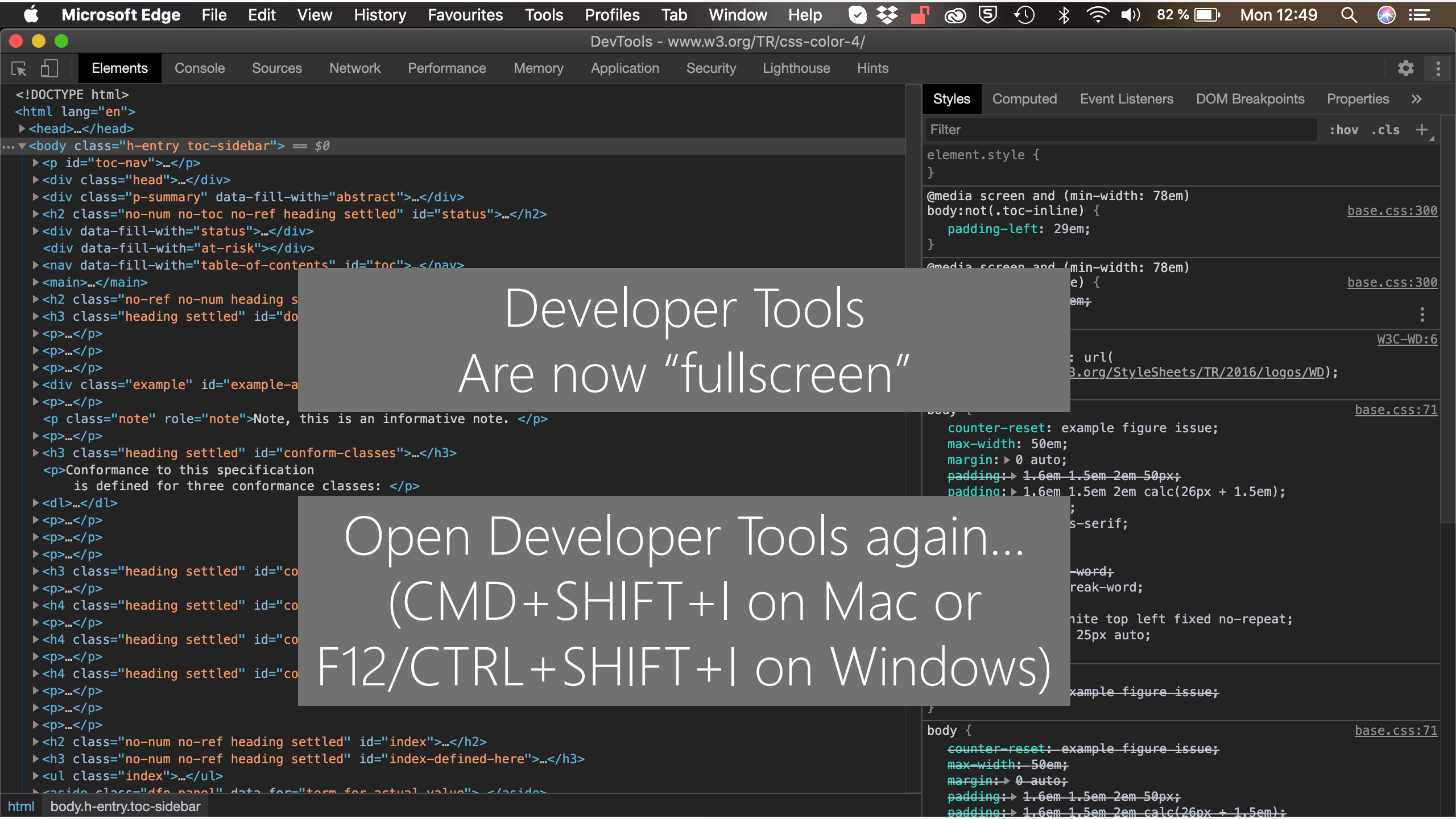
Activate the ... menu and
 Select "Undock into separate window"

Properties

```

font-size: 29em;
font-family: sans-serif;

```



Developer Tools
Are now "fullscreen"

Open Developer Tools again...
(CMD+SHIFT+I on Mac or
F12/CTRL+SHIFT+I on Windows)

```

<!DOCTYPE html>
<html lang="en">
  <head>...</head>
  <body class="h-entry toc-sidebar"> == $0
    <p id="toc-nav">...</p>
    <div class="head">...</div>
    <div class="p-summary" data-fill-with="abstract">...</div>
    <h2 class="no-num no-toc no-ref heading settled" id="status">...</h2>
    <div data-fill-with="status">...</div>
    <div data-fill-with="at-risk">...</div>
    <nav data-fill-with="table-of-contents" id="toc">...</nav>
    <main>...</main>
    <h2 class="no-ref no-num heading settled" id="conformance">...</h2>
    <h3 class="heading settled" id="document-conventions">...</h3>
    <p>...</p>
    <p>...</p>
    <p>...</p>
    <div class="example" id="example-ae2">...</div>
    <p>...</p>
    <p class="note" role="note">Note, th
    <p>...</p>
    <h3 class="heading settled" id="conf
    <p>Conformance to this specification
    is defined for three conformance
    <dl>...</dl>
    <p>...</p>
    <p>...</p>
    <p>...</p>
    <h3 class="heading settled" id="conf
    <p>...</p>
    <h4 class="heading settled" id="conf
    <p>...</p>
    <h4 class="heading settled" id="conf
    <p>...</p>
    <h4 class="heading settled" id="conf
    <p>...</p>
    <p>...</p>
    <p>...</p>
    <p>...</p>
    <h2 class="no-num no-ref heading settled" id="index">...</h2>
    <h3 class="no-num no-ref heading settled" id="index-defined-here">...</h3>
    <ul class="index">...</ul>
    <aside class="dfn-panel" data-for="term-for-actual-value">...</aside>
  
```

```

DevTools - devtools://devtools/bundled/devtools_app.html?remoteBase=https://devtools.azureedge.net/serve_file/@34...
Elements Console Sources Network Performance Memory Application Security Lighthouse
top Filter Default levels
Main._createAppUI: 50.18603515625ms main.js:31
Main._showAppUI: 78.2470703125ms main.js:31
Main._initializeTarget: 54.614990234375ms main.js:31
Main._lateInitialization: 4.332763671875ms main.js:31
>

```

You now can debug the developer tools with developer tools (I changed to light mode to make it obvious which are which)

while(🐢) {go(deeper)}

Elements Console Sources Network Performance Memory Application Security Lighthouse

span.tabbed-pane-header-tab-titl 44.23 x 15 e

Elements Console Sources Network Performance Memory Application Security Lighthouse

```

<div class="tabbed-pane-header-tabs" role="tablist" aria-label="Panels" style>
  ><div class="tabbed-pane-header-tab" id="tab-elements" role="tab" aria-selected="false" aria-label="Elements" style="cursor: pointer; width: 73px;">...</div>
  ><div class="tabbed-pane-header-tab selected" id="tab-console" role="tab" aria-selected="true" aria-label="Console" style="cursor: pointer; width: 68px; tabindex="0">
    <span class="tabbed-pane-header-tab-title">Console</span> == $0
  </div>
  ><div class="tabbed-pane-header-tab" id="tab-sources" role="tab" aria-selected="false" aria-label="Sources" style="cursor: pointer; width: 68px;">...</div>
  ><div class="tabbed-pane-header-tab" id="tab-network" role="tab" aria-selected="false" aria-label="Network" style="cursor: pointer; width: 69px;">...</div>
  ><div class="tabbed-pane-header-tab" id="tab-timeline" role="tab" aria-selected="false" aria-label="Performance" style="cursor: pointer; width: 93px;">...</div>
  ><div class="tabbed-pane-header-tab" id="tab-heap_profiler" role="tab" aria-selected="false" aria-label="Memory" style="cursor: pointer; width: 68px;">...</div>
  ><div class="tabbed-pane-header-tab" id="tab-resources" role="tab" aria-selected="false" aria-label="Application" style="cursor: pointer; width: 84px;">...</div>
  ><div class="tabbed-pane-header-tab" id="tab-security" role="tab" aria-selected="false" aria-label="Security" style="cursor: pointer; width: 67px;">...</div>
  ><div class="tabbed-pane-header-tab" id="tab-lighthouse" role="tab" aria-selected="false" aria-label="Lighthouse" style="cursor: pointer; width: 83px;">...</div>
</div>
<div class="tabbed-pane-content">...</div>
</div>
<div class="widget vbox flex-auto view-container overflow-auto" tabindex="-1" role="tabpanel" aria-label="Console panel">...</div>
</div>
</div>
</div>

```

Styles >>

```

:hov .cls
element.style {
}

```

```

ui/tabbedP...
.tabbed-pane-header-tab-title {
  text-overflow: ellipsis;
  overflow: hidden;
}

```

```

ui/inspect...
* {
  box-sizing: border-box;
}

```

```

ui/inspect...
* {
  min-width: 0;
  min-height: 0;
}

```

Inherited from...

```

Style Attribute {
  cursor: pointer;
  width: 68px;
}

```

tabbedPane... .tabbed-pane-header-

... div div div div div #shadow-root div div div div #tab-console span.tabbed-pane-header-tab-title

Lots of fun bits to discover (and report if you find a problem)



In-built reporting tool...



Sources >> [Sources icon] [Profile icon] [More icon]

or: rgb(21, 32, 43); overflow-y:

Click to give Chris more work, please add an email to follow up

margin -

What makes me happy?





What are developer tools made of?



- I like that we use web technology to build developer tools
- This makes them embeddable into other software (VS Code is an Electron shell with Monaco inside)
- It also allows us to build for the web on the web (CodePen, JSBin, Code Sandbox, Workspaces for GitHub)
- It is fun to work in them as you have a predictable environment – we can test upcoming web tech there.

Little touches that go a long way...



```
const doFormStuff = (ev) => {  
  
};
```

Little touches that go a long way...

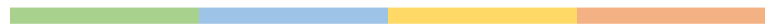
```
const doFormStuff = (ev) => {  
  
};
```

```
const doFormStuff = (ev) => {  
  
};
```

```
document.querySelector('form').  
  |  addEventListener('submit', doFormStuff);
```



Webhint: testing and best practices



- Testing tool to check web projects for:
 - Accessibility issues
 - Performance problems
 - App readiness
 - Compatibility with standards
 - Common pitfalls to avoid
 - Security issues
- Online service (webhint.io) or node package to integrate in other tools
- Highly customisable



Webhint in VS Code



```
logogen.js — logo-o-matic
JS fonts.js JS logogen.js X index.html rip.html # logogen.css n
js > JS logogen.js > <function>
62 white: [255, 255, 255, 255],
63 red: [104, 55, 43, 255],
64 cyan: [112, 164, 178, 255],
65 purple: [111, 61, 134, 255],
66 green: [77, 141, 67, 255],
67 blue: [53, 40, 121, 255],
68 yellow: [184, 199, 111, 255],
69 orange: [111, 79, 37, 255],
70 brown: [67, 57, 0, 255],
71 lightred: [154, 103, 89, 255],
72 darkgrey: [68, 68, 68, 255],
73 grey: [108, 108, 108, 255],
74 lightgreen: [154, 210, 132, 255],
75 lightblue: [108, 94, 181, 255],
76 lightgrey: [149, 149, 149, 255]

PROBLEMS 2 OUTPUT DEBUG CONSOLE ... Filter: E.g.: text, **/*.ts, !**/node_modules/**
rip.html 1
  x ^ 'viewport' meta element was not specified.
    (meta-viewport) webhint [1, 1]
index.html 1
  ! ^ 'a[download]' is not supported by Internet Explorer, iOS Safari.
    (compat-api/html) webhint [88, 6]
```

JS fonts.js

<> index.html >

81

82

83

84

85

86

87

88

89

90

91

92

<a>

If the a element has an href attribute, then it represents a hyperlink (a hypertext anchor) labeled by its contents.

[MDN Reference](#)

'a[download]' is not supported by Internet Explorer, iOS Safari.

(compat-api/html) webhint

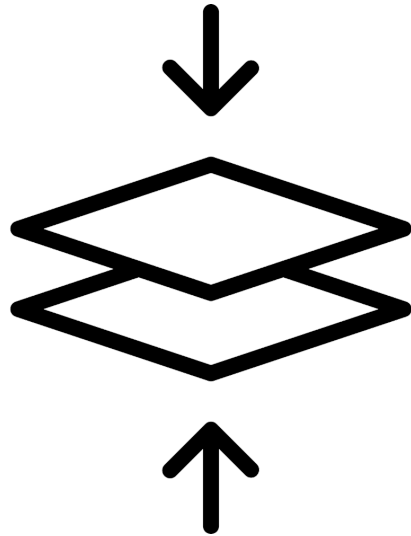
< Peek Problem (⌘F8) No quick fixes available

```
<a id="save" href="" download="">
```

```
<svg height='100%' width='100%' fill="#ffffff"  
viewBox="0 0 847 847" x="0px" y="0px" aria-labelledby="s
```

```
<title id="savetitle">Save</title>
```

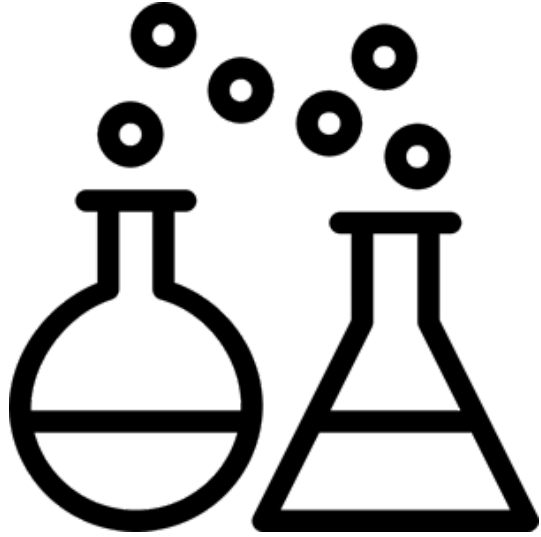
```
<g>
```



Merging of tools



- VS Code is not only an editor, it also includes version control and a terminal
- This means I don't have to jump from one context to another to debug
- However... wouldn't it be cool to have Devtools in VS Code?



Devtools for Code

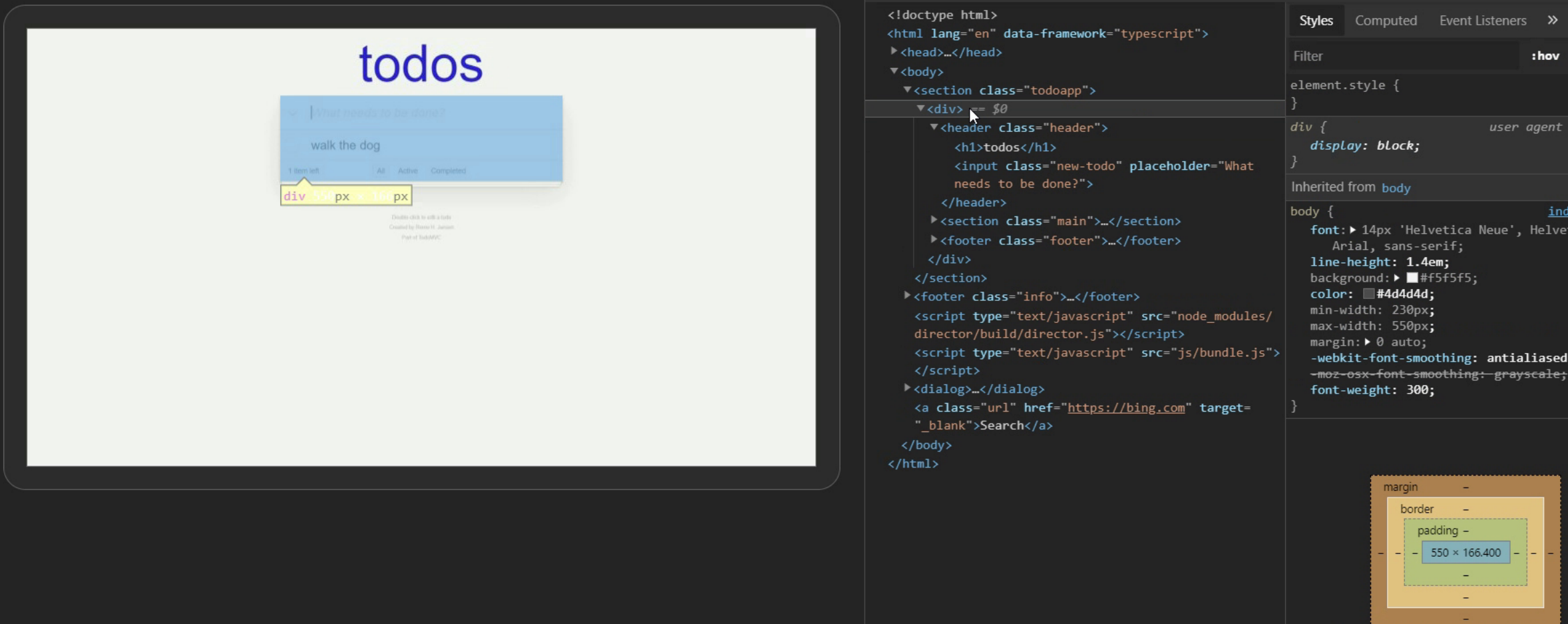


- Edge/Chromium developer tools inside VS Code
- No need to switch to the browser to try a few things out
- No need to replicate the same functionality in both places

aka.ms/elements4code

Elements x index.scss

localhost:4200/#/



```

<!doctype html>
<html lang="en" data-framework="typescript">
  <head>...</head>
  <body>
    <section class="todoapp">
      <div class="header">
        <h1>todos</h1>
        <input class="new-todo" placeholder="What needs to be done?">
      </div>
      <section class="main">...</section>
      <footer class="footer">...</footer>
    </section>
    <footer class="info">...</footer>
    <script type="text/javascript" src="node_modules/director/build/director.js"></script>
    <script type="text/javascript" src="js/bundle.js"></script>
    <dialog>...</dialog>
    <a class="url" href="https://bing.com" target="_blank">Search</a>
  </body>
</html>

```

Styles Computed Event Listeners >>

Filter :hov .cls +

element.style { }

div { user agent stylesheet }

display: block;

Inherited from body

body { index.scss:2 }

font: 14px 'Helvetica Neue', Helvetica, Arial, sans-serif;

line-height: 1.4em;

background-color: #f5f5f5;

color: #4d4d4d;

min-width: 230px;

max-width: 550px;

margin: 0 auto;

-webkit-font-smoothing: antialiased;

-moz-osx-font-smoothing: grayscale;

font-weight: 300;

margin - - - -

border - - - -

padding - - - -

550 x 166.400

aka.ms/elements4code

Thanks – your voice matters!



Chris Heilmann

@codepo8

@EdgeDevTools

christianheilmann.com

Click the feedback in Edge developer tools!

