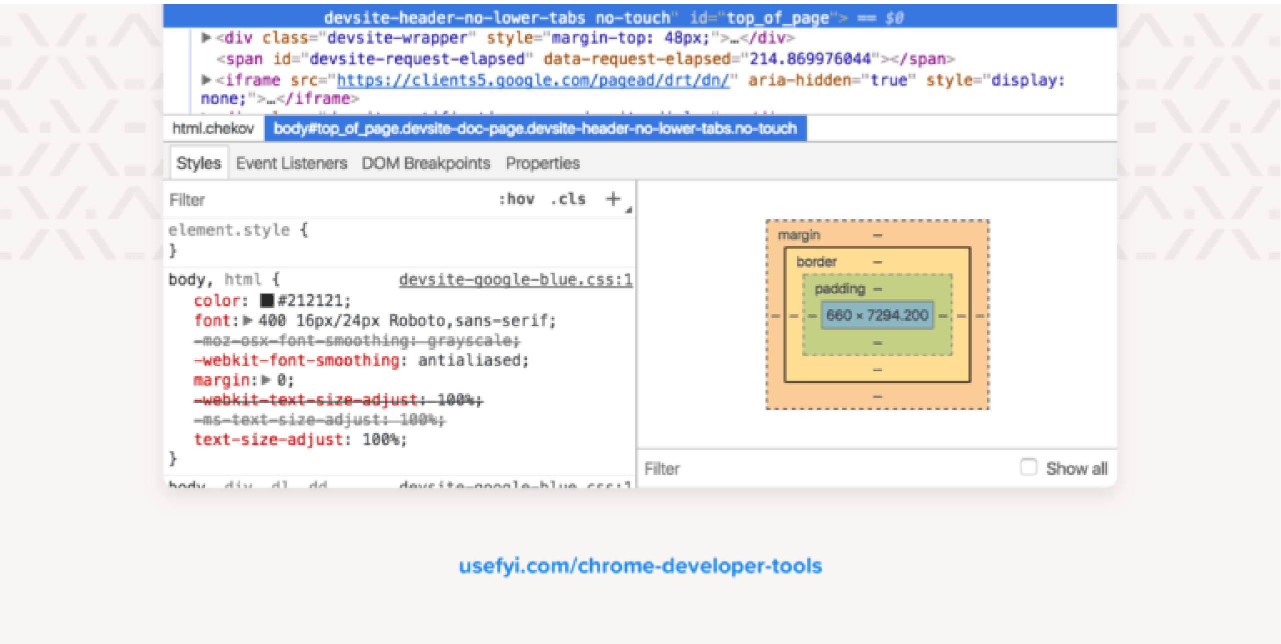
nira Blog

The Beginner’s Guide to Chrome Developer Tools

Marie Prokopets,
Co-founder of Nira



```
devsite-header-no-lower-tabs no-touch" id="top_of_page"> == $0
  ▶<div class="devsite-wrapper" style="margin-top: 48px;">…</div>
    <span id="devsite-request-elapsed" data-request-elapsed="214.869976044"></span>
  ▶<iframe src="https://clients5.google.com/pagead/drt/dn/" aria-hidden="true" style="display:
  none;">…</iframe>
```

html.chekov body#top_of_page.devsite-doc-page.devsite-header-no-lower-tabs.no-touch

```
element.style {
}
body, html {                    devsite-google-blue.css:1
   color: ■#212121;
   font:▶ 400 16px/24px Roboto,sans-serif;
   ~~-moz-osx-font-smoothing: grayscale;~~
   -webkit-font-smoothing: antialiased;
   margin:▶ 0;
   ~~-webkit-text-size-adjust: 100%;~~
   ~~-ms-text-size-adjust: 100%;~~
   text-size-adjust: 100%;
}
```

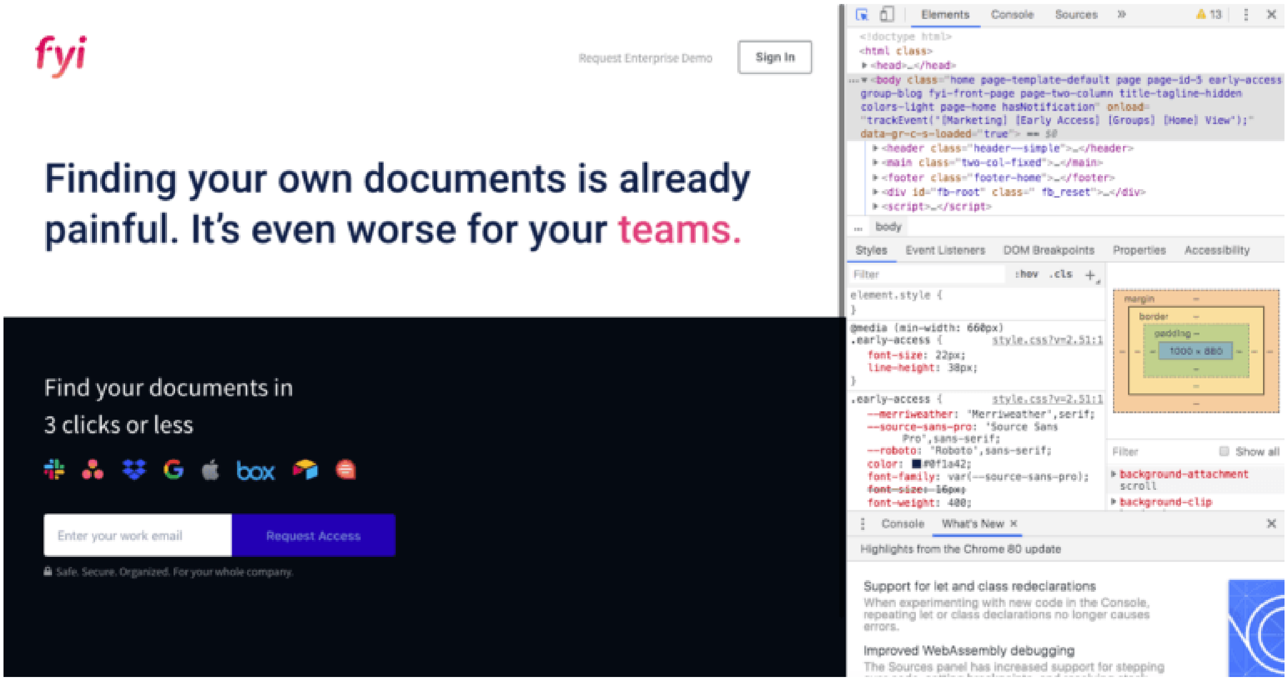
usefyi.com/chrome-developer-tools

Chrome Developer Tools is a comprehensive toolkit for developers, built directly into the Chrome browser. These tools let you edit web pages in real time, diagnose problems more quickly, and build better websites faster.

Even if you’re not a developer, though, you can still get some use out of Developer Tools.

What are Chrome Developer Tools?

Here’s what you’ll see when you open Developer Tools:



Finding your own documents is already painful. It’s even worse for your teams.

Find your documents in 3 clicks or less

Enter your work email

Request Access

Safe. Secure. Organized. For your whole company.

```
<!doctype html>
<html class>
▶<head>…</head>
…▼<body class="home page-template-default page page-id-5 early-access
  group-blog fyi-front-page page-two-column title-tagline-hidden
  colors-light page-home hasNotification" onload=
  "trackEvent("[Marketing] [Early Access] [Groups] [Home] View");"
  data-gr-c-s-loaded="true"> == $0
  ▶<header class="header--simple">…</header>
  ▶<main class="two-col-fixed">…</main>
  ▶<footer class="footer-home">…</footer>
  ▶<div id="fb-root" class=" fb_reset">…</div>
  ▶<script>…</script>
```

```
element.style {
}
@media (min-width: 660px)
.early-access {           style.css?v=2.51:1
    font-size: 22px;
    line-height: 38px;
}
.early-access {           style.css?v=2.51:1
    --merriweather: 'Merriweather',serif;
    --source-sans-pro: 'Source Sans
        Pro',sans-serif;
    --roboto: 'Roboto',sans-serif;
    color: ■#0f1a42;
    font-family: var(--source-sans-pro);
    ~~font-size: 16px;~~
    font-weight: 400;
```

```
▶ background-attachment
  scroll
▶ background-clip
```

Highlights from the Chrome 80 update

Support for let and class redeclarations

When experimenting with new code in the Console, repeating let or class declarations no longer causes errors.

Improved WebAssembly debugging

The Sources panel has increased support for stepping

A standard web page, but with a new set of tools open in a panel on the right of the page.

There are three tabs at the top of the Developer Tools window, and a further six that you can see by clicking the **>>** symbol next to them.

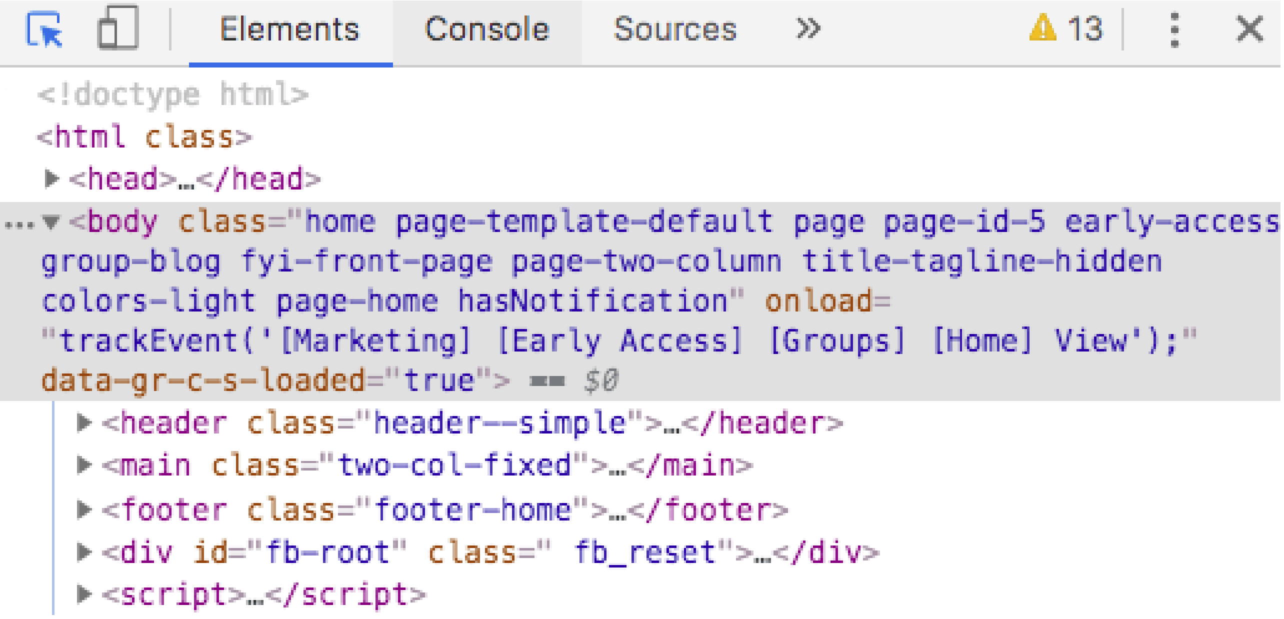
The tabs are:

- Elements
- Console
- Source
- Network
- Application
- Security
- Memory
- Performance
- Audits

They don’t always stay in this order—Chrome moves them around based on which you opened last.

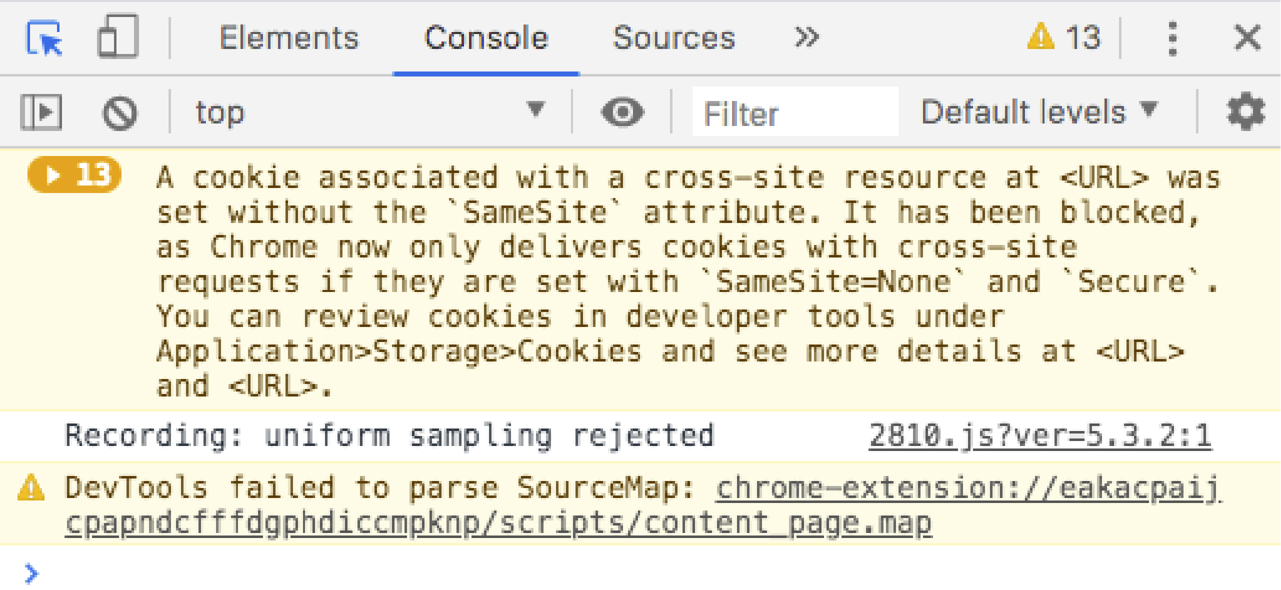
By default, Developer Tools opens with the Elements tab open.

Elements shows you the HTML used to build the page you're looking at, together with any inline CSS.



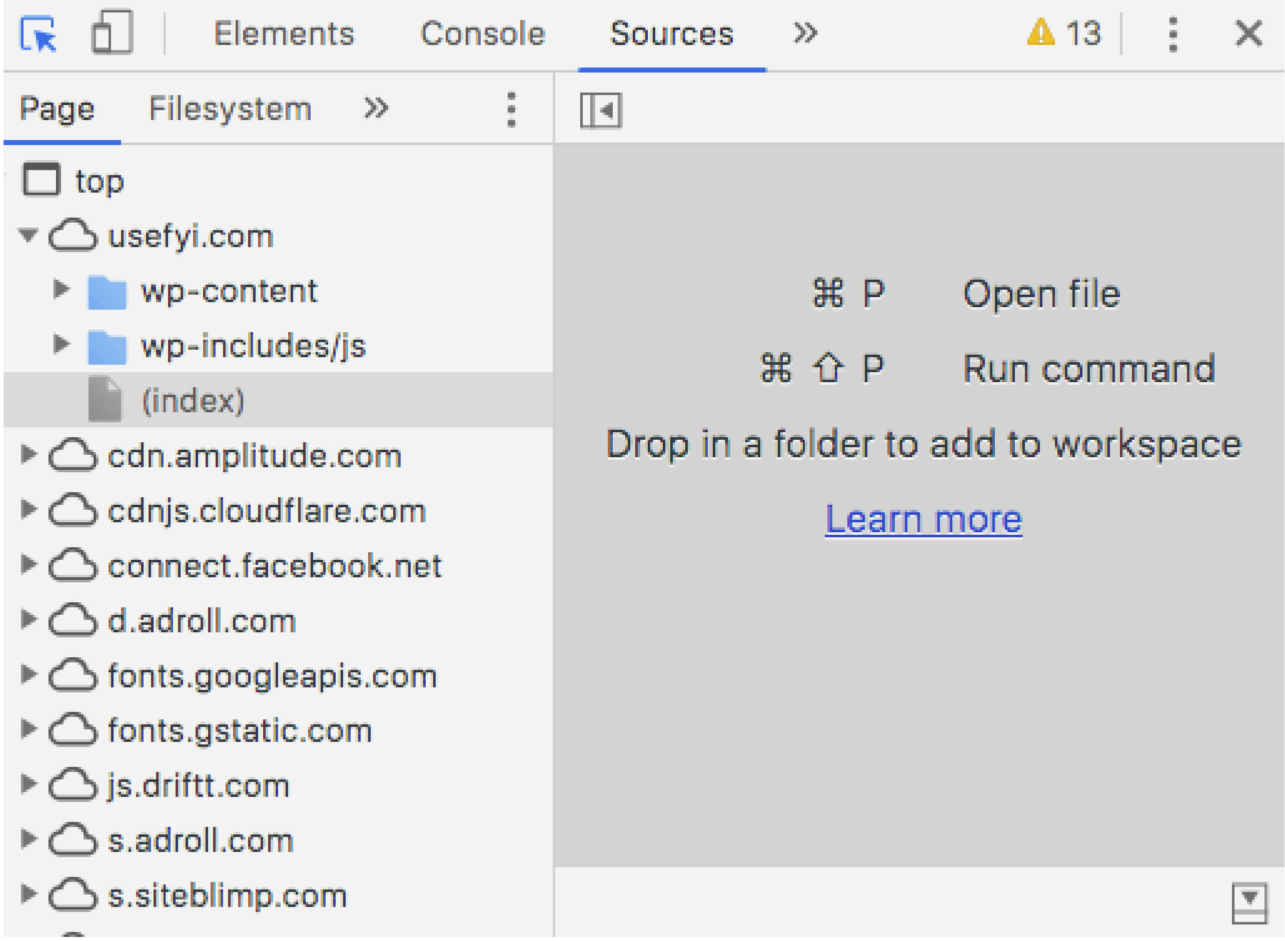
```
<!doctype html>
<html class>
 ▶<head>…</head>
…▼<body class="home page-template-default page page-id-5 early-access
  group-blog fyi-front-page page-two-column title-tagline-hidden
  colors-light page-home hasNotification" onload=
  "trackEvent('[Marketing] [Early Access] [Groups] [Home] View');"
  data-gr-c-s-loaded="true"> == $0
   ▶<header class="header--simple">…</header>
   ▶<main class="two-col-fixed">…</main>
   ▶<footer class="footer-home">…</footer>
   ▶<div id="fb-root" class=" fb_reset">…</div>
   ▶<script>…</script>
```

Console deals with JavaScript. It gives you information about interactive elements on a page. In Console, you can write JavaScript to interact with the web page you're viewing, and it also lets you write messages to yourself in the JavaScript of websites you're building, which then show up in Console to show that the JS was executed.

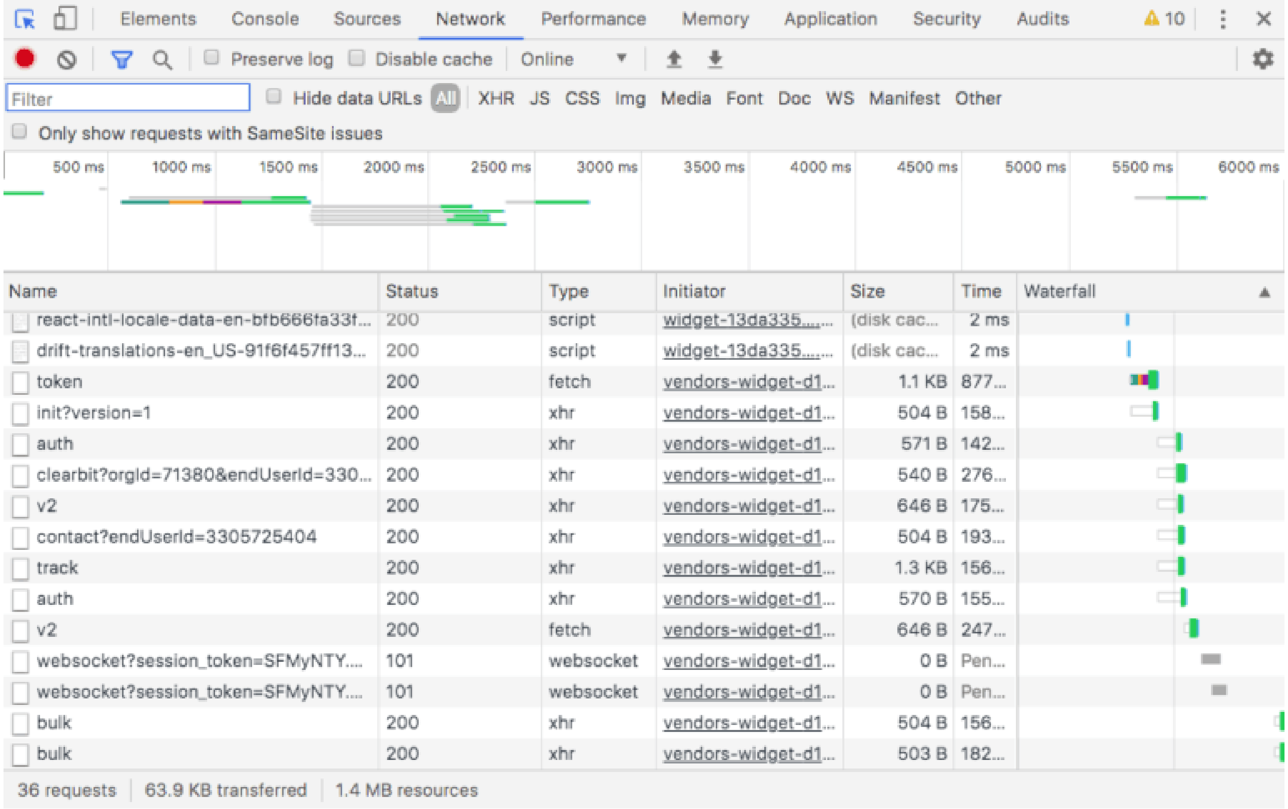


```
▶ 13  A cookie associated with a cross-site resource at <URL> was
      set without the `SameSite` attribute. It has been blocked,
      as Chrome now only delivers cookies with cross-site
      requests if they are set with `SameSite=None` and `Secure`.
      You can review cookies in developer tools under
      Application>Storage>Cookies and see more details at <URL>
      and <URL>.
   Recording: uniform sampling rejected          2810.js?ver=5.3.2:1
⚠  DevTools failed to parse SourceMap: chrome-extension://eakacpaij
   cpapndcfffdgphdiccmpknp/scripts/content_page.map
›
```

The **Sources** tab shows you where all the files that were used to make the website are stored and lets you inspect them.



The **Network** tab shows you all the files that are loading in the URL you’re looking at.

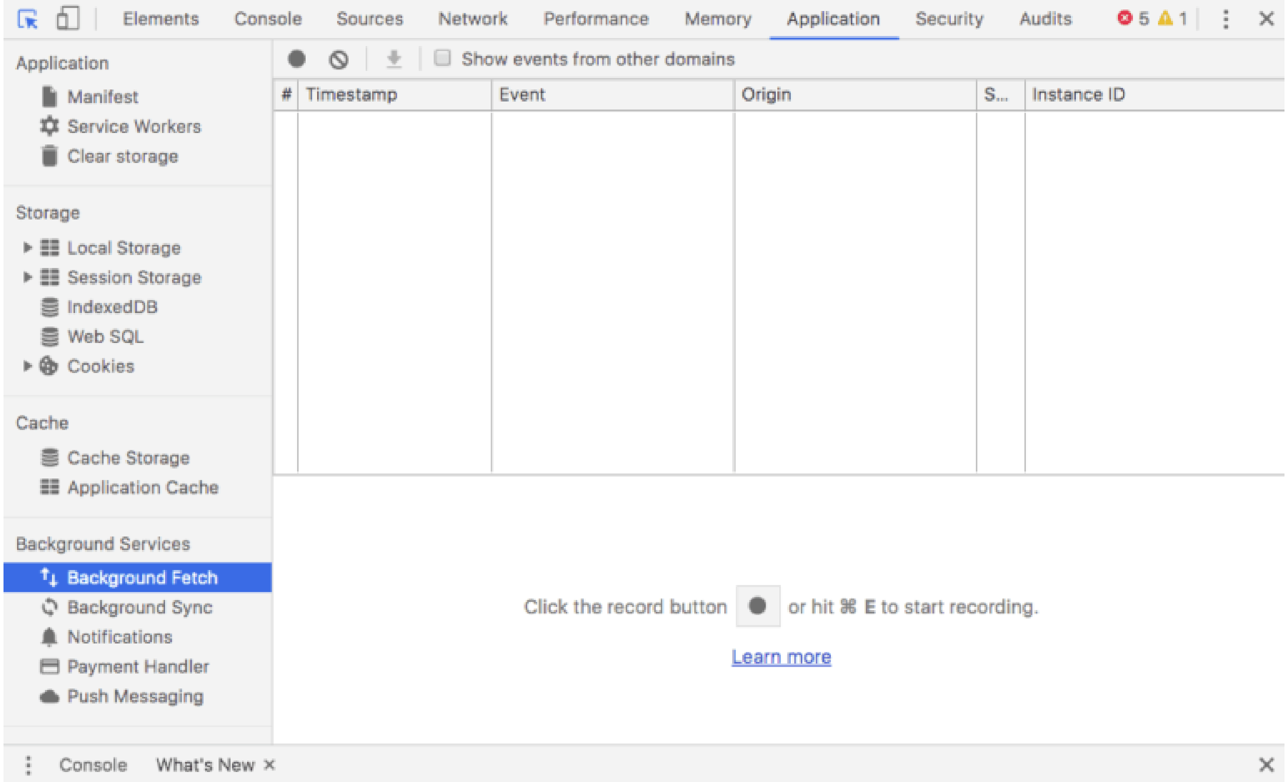


Name	Status	Type	Initiator	Size	Time	Waterfall
react-intl-locale-data-en-bfb666fa33f...	200	script	widget-13da335......	(disk cac...	2 ms	
drift-translations-en_US-91f6f457ff13...	200	script	widget-13da335......	(disk cac...	2 ms	
token	200	fetch	vendors-widget-d1...	1.1 KB	877...	
init?version=1	200	xhr	vendors-widget-d1...	504 B	158...	
auth	200	xhr	vendors-widget-d1...	571 B	142...	
clearbit?orgId=71380&endUserId=330...	200	xhr	vendors-widget-d1...	540 B	276...	
v2	200	xhr	vendors-widget-d1...	646 B	175...	
contact?endUserId=3305725404	200	xhr	vendors-widget-d1...	504 B	193...	
track	200	xhr	vendors-widget-d1...	1.3 KB	156...	
auth	200	xhr	vendors-widget-d1...	570 B	155...	
v2	200	fetch	vendors-widget-d1...	646 B	247...	
websocket?session_token=SFMyNTY....	101	websocket	vendors-widget-d1...	0 B	Pen...	
websocket?session_token=SFMyNTY....	101	websocket	vendors-widget-d1...	0 B	Pen...	
bulk	200	xhr	vendors-widget-d1...	504 B	156...	
bulk	200	xhr	vendors-widget-d1...	503 B	182...	

36 requests | 63.9 KB transferred | 1.4 MB resources

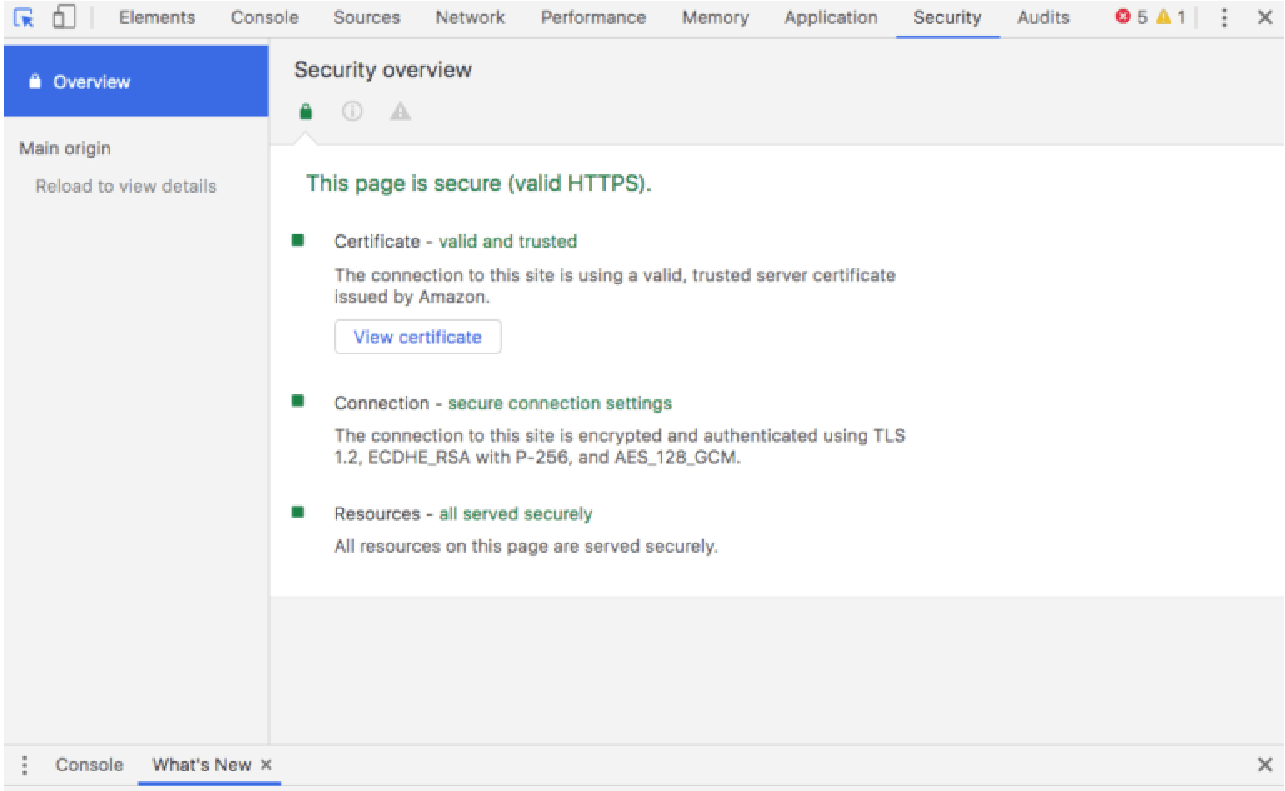
You get a waterfall and deep data on all the items loaded, including initiator and time to load that element.

Application shows you what’s in your browser storage: in-browser databases like Web SQL, local storage, and more. It also gives you granular control over your cookies.



Click the record button ● or hit ⌘ E to start recording.

Security gives you basic security information, letting you view a site’s HTTPS certificate and TLS status.



Security overview

This page is secure (valid HTTPS).

Certificate - valid and trusted

The connection to this site is using a valid, trusted server certificate issued by Amazon.

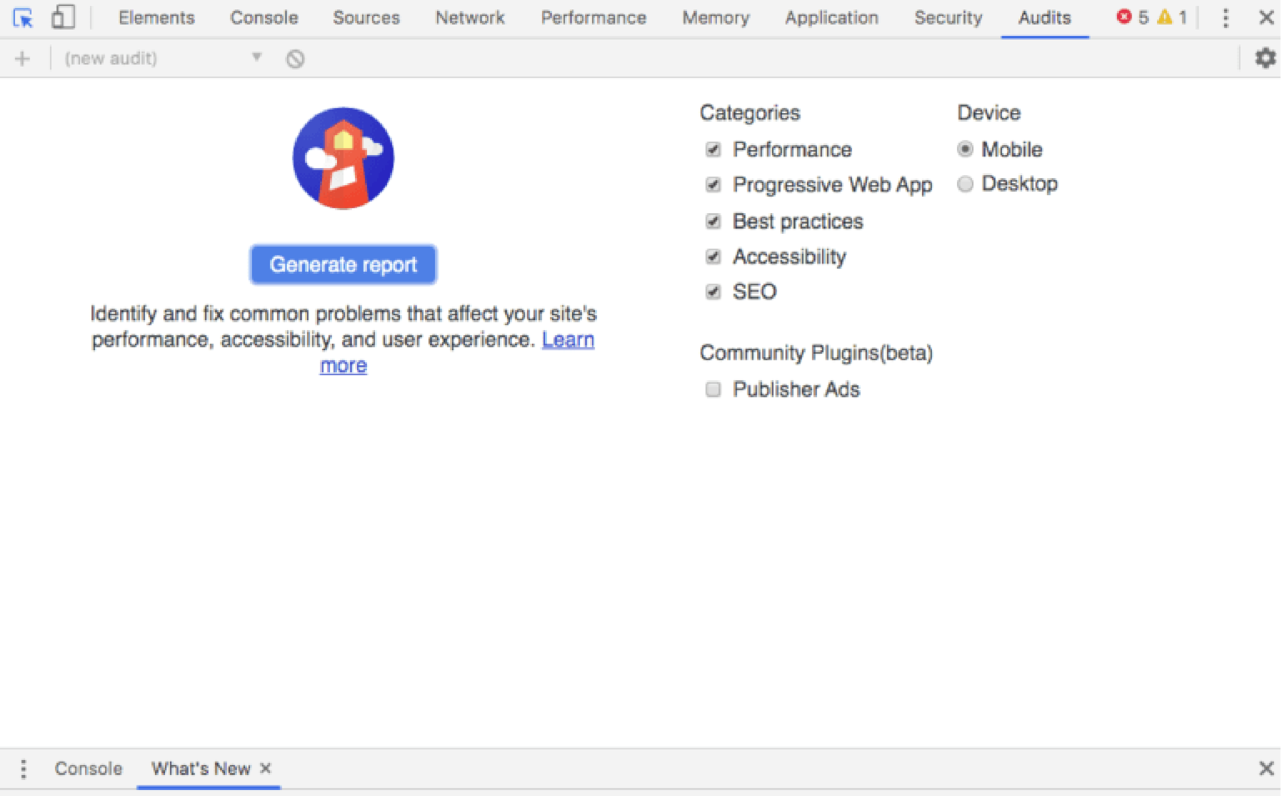
Connection - secure connection settings

The connection to this site is encrypted and authenticated using TLS 1.2, ECDHE_RSA with P-256, and AES_128_GCM.

Resources - all served securely

All resources on this page are served securely.

Finally, there’s **Audits**, which autogenerates reports on site functionality and structure to help developers improve performance.



Identify and fix common problems that affect your site's performance, accessibility, and user experience. Learn more

Categories

☑ Performance
☑ Progressive Web App
☑ Best practices
☑ Accessibility
☑ SEO

Community Plugins(beta)

□ Publisher Ads

Device

● Mobile
○ Desktop

Developer Tools is updated with each new Chrome release. To make sure they’re up-to-date, go to Components by typing **chrome://components** into your omnibar, select “Recovery” and check for updates.

Important: You can’t change the code on the website with Developer Tools. You’d need access to the site’s backend for that. Only the code your browser uses to display the website is changed. So even if you’re a total beginner, you can do anything you like with Developer Tools with no real risk. Close the window, go back to the website, and everything is the same as it was before.

Let’s start using Developer Tools.

How to open Chrome Developer Tools

You can open Developer Tools with keyboard shortcuts or through the Chrome menu.

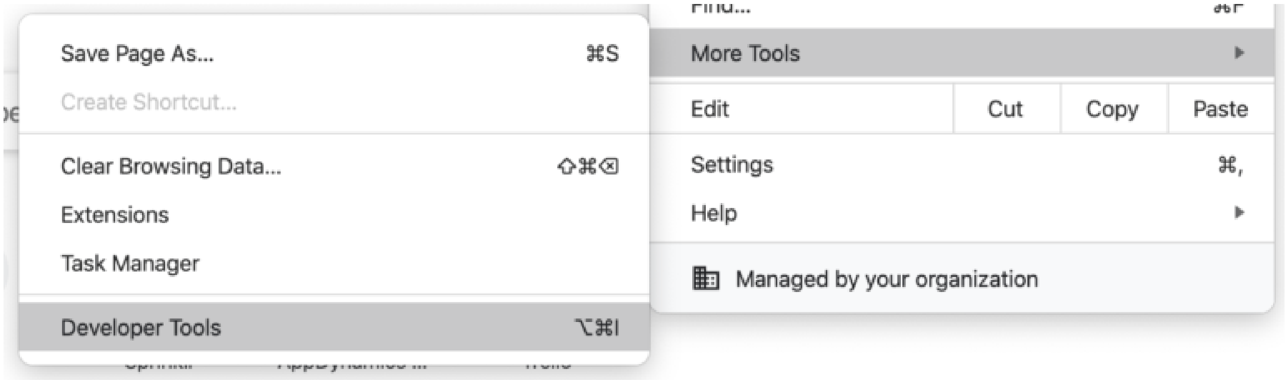
The keyboard shortcuts are:

- Mac OS: **CMD+Shift+J** or **CMD+Shift+C**

- Linux, Chromebook and Windows: **Ctrl+Shift+J**

From the Chrome menu:

Open the Chrome menu and go to "More Tools" > "Developer Tools."



Finally, you can right-click (Windows) or Ctrl-click (Mac) anything on a web page and select "Inspect Element" to open Developer Tools.

The Developer Tools panel will open in whatever web page you're on. You need to open them again for each new Chrome window, but you can navigate between web pages with them open.

What about on mobile?

Opening Developer Tools on mobile

You can't access Developer Tools directly on Android. Instead you'll need to use Remote Debugging. The basic process works like this:

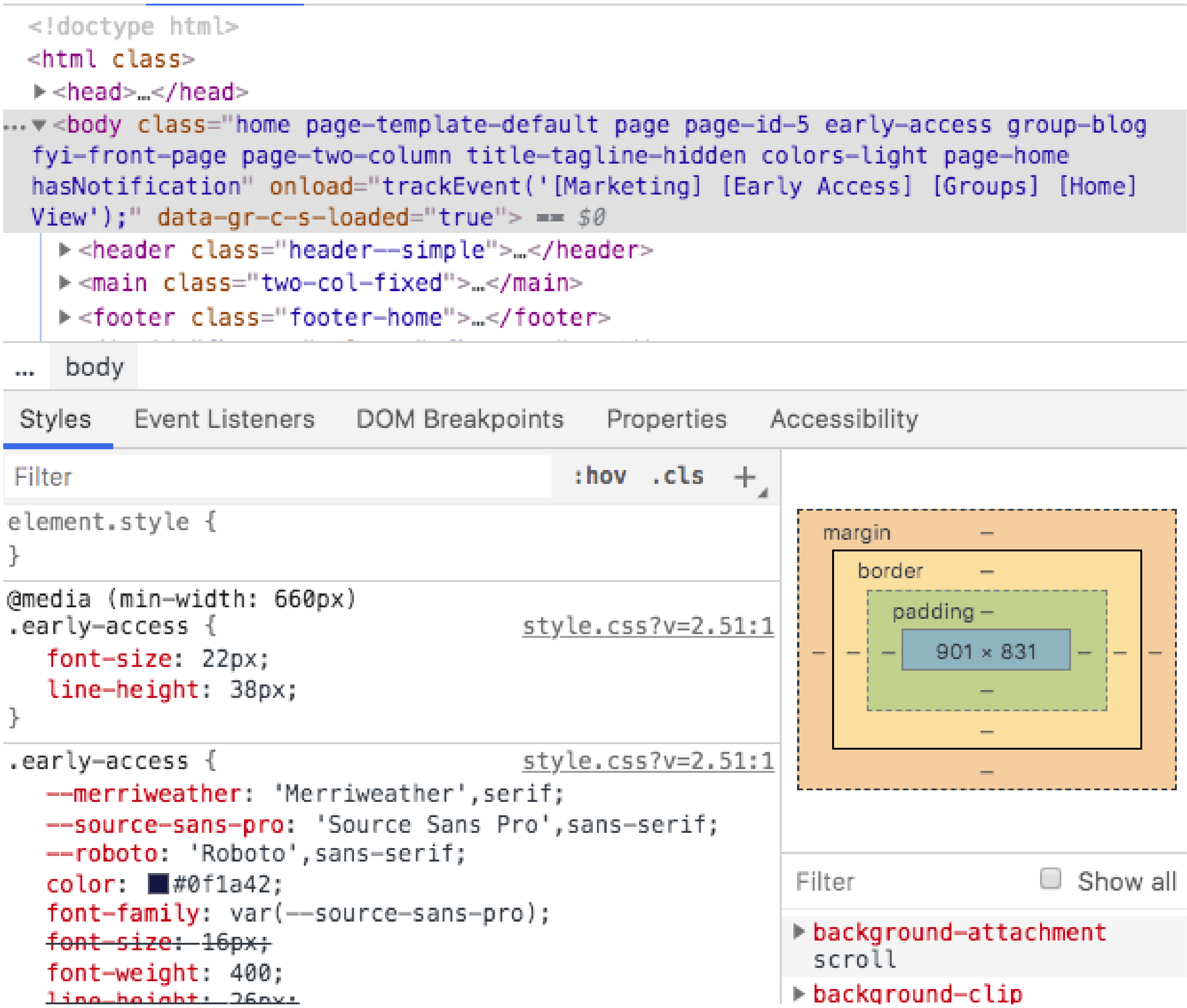
1. Connect your Android device
2. Open Developer Tools and go to "More Tools" > "Remote Devices"
3. Authorize the remote access on your Android device

There doesn't seem to be a way to do this at all on iOS devices.

5 common uses of Chrome Developer Tools for beginners

Just because you're not a developer doesn't mean Developer Tools aren't for you. Here are five ways you can make your life easier with Developer Tools.

Most of these will use the Elements tab. Here's a quick guide to that:



```
<!doctype html>
<html class>
 ▶<head>…</head>
…▼<body class="home page-template-default page page-id-5 early-access group-blog
  fyi-front-page page-two-column title-tagline-hidden colors-light page-home
  hasNotification" onload="trackEvent('[Marketing] [Early Access] [Groups] [Home]
  View');" data-gr-c-s-loaded="true"> == $0
   ▶<header class="header--simple">…</header>
   ▶<main class="two-col-fixed">…</main>
   ▶<footer class="footer-home">…</footer>
```

```
element.style {
}
@media (min-width: 660px)
.early-access {                          style.css?v=2.51:1
   font-size: 22px;
   line-height: 38px;
}
.early-access {                          style.css?v=2.51:1
   --merriweather: 'Merriweather',serif;
   --source-sans-pro: 'Source Sans Pro',sans-serif;
   --roboto: 'Roboto',sans-serif;
   color: ■#0f1a42;
   font-family: var(--source-sans-pro);
   ~~font-size: 16px;~~
   font-weight: 400;
   ~~line-height: 26px;~~
```

The top section shows the HTML for the site.

The bottom left section shows the CSS styles applied to that page.

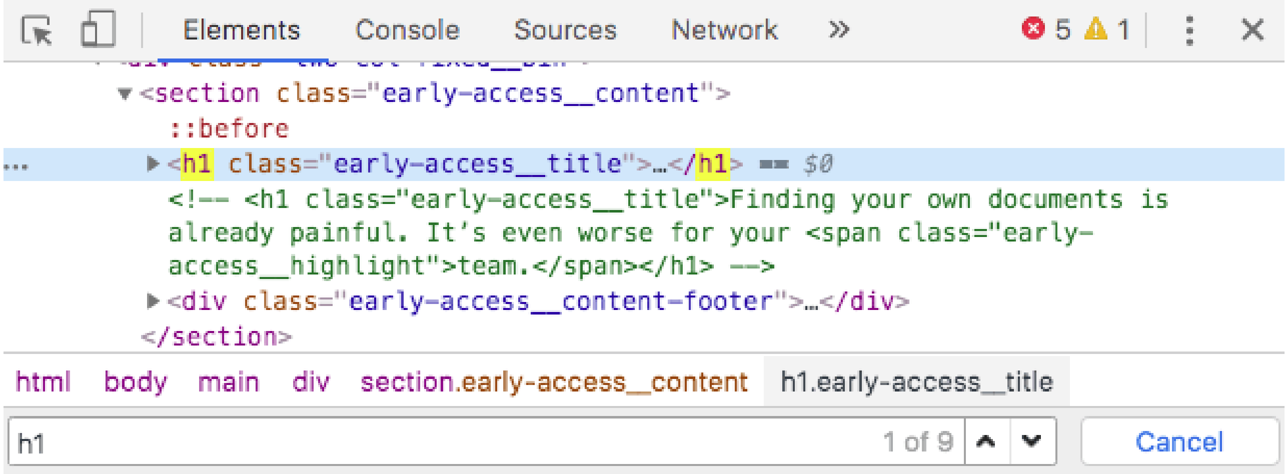
The bottom right shows the CSS layout visually so you know what you're looking at.

1. Inspect page tags

One of the most important elements of onpage SEO is simple h1 and h2 tags. Snoop under the hood and you'll find a lot of folks (including those who should know better) aren't getting this stuff right.

Once you've seen pages where the h1 was an unoptimized image, or pages with four h1 tags, you begin to appreciate the urgency of the situation. And tools like Screaming Frog are great but they're not always 100% accurate. It's best to check by hand. It's also very easy.

Open a page, then open Developer Tools. In the Elements tab, press **CMD+F** or **Ctrl-F** to search and search "h1" to see your page's tag(s):



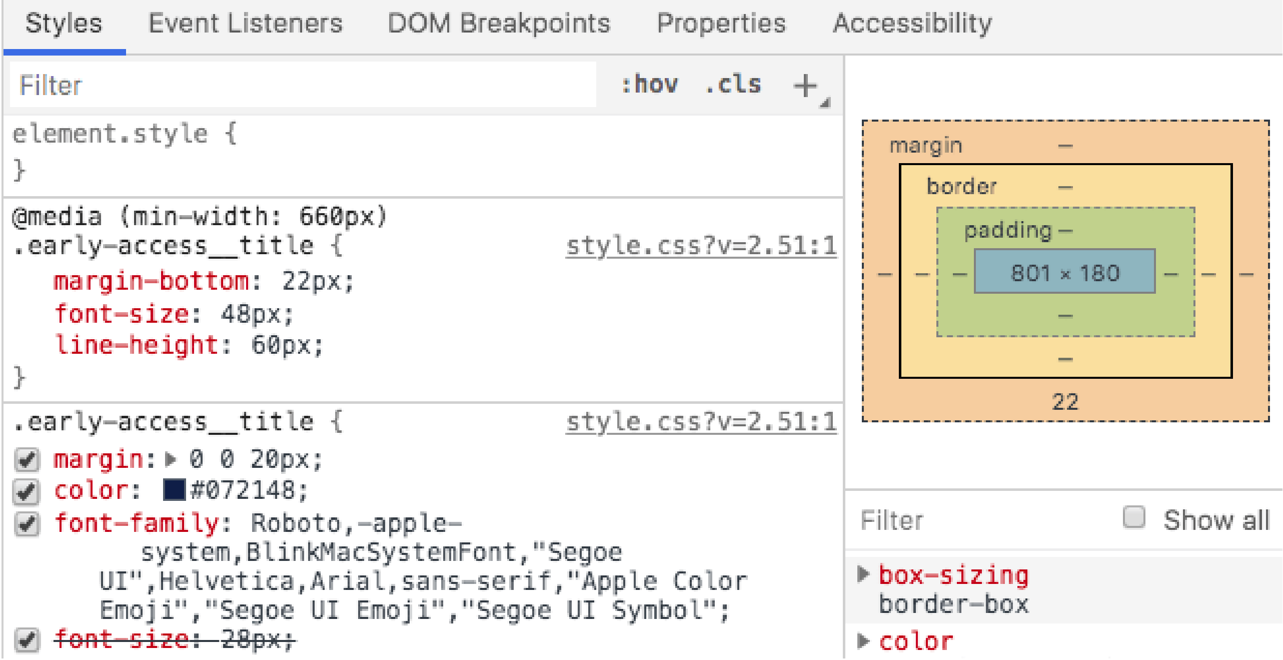
```
      ▼<section class="early-access__content">
         ::before
...    ▶<h1 class="early-access__title">…</h1> == $0
         <!-- <h1 class="early-access__title">Finding your own documents is
         already painful. It's even worse for your <span class="early-
         access__highlight">team.</span></h1> -->
       ▶<div class="early-access__content-footer">…</div>
      </section>
```

2. Edit CSS

Developer Tools lets you change the CSS of a page directly in your browser. You can trial color combinations, fonts—anything that's governed by CSS. You don't have to mess around with the stylesheet and you often don't have to write a single character of code.

Suppose I want to see what the FYI homepage would look like with red text.

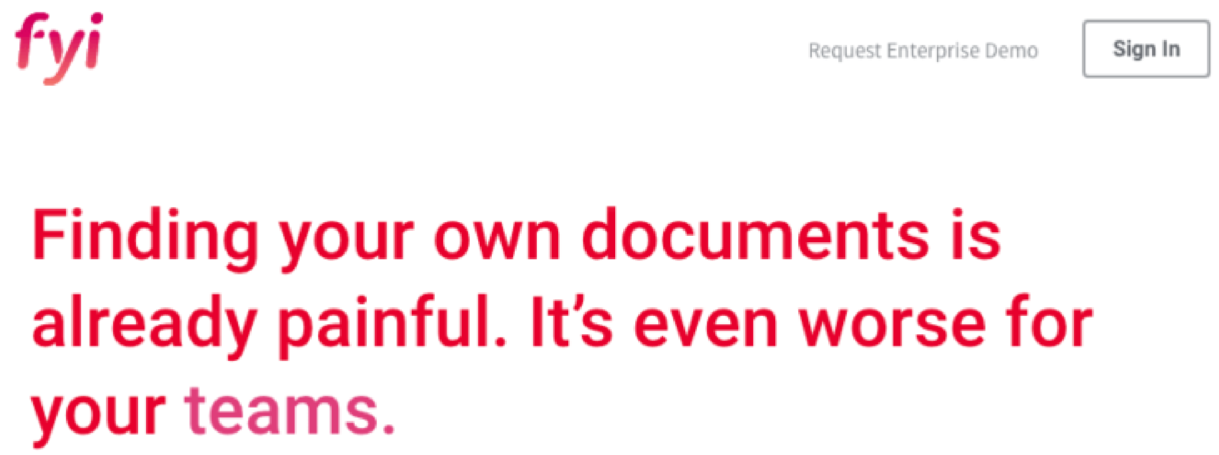
I open the page, then open Developer Tools. Ctrl-click or right-click the element you want to change on the web page. In the Elements tab I find the color for the element in the bottom left panel:



```
element.style {
}
@media (min-width: 660px)
.early-access__title {                    style.css?v=2.51:1
   margin-bottom: 22px;
   font-size: 48px;
   line-height: 60px;
}
.early-access__title {                    style.css?v=2.51:1
☑  margin: ▶ 0 0 20px;
☑  color: ■#072148;
☑  font-family: Roboto,-apple-
         system,BlinkMacSystemFont,"Segoe
      UI",Helvetica,Arial,sans-serif,"Apple Color
      Emoji","Segoe UI Emoji","Segoe UI Symbol";
☑  ~~font-size: 28px;~~
```

```
▶ box-sizing
  border-box
▶ color
```

Click on the color box and I can simply click on another color in Chrome Color Picker (more on this later). The text changes immediately.



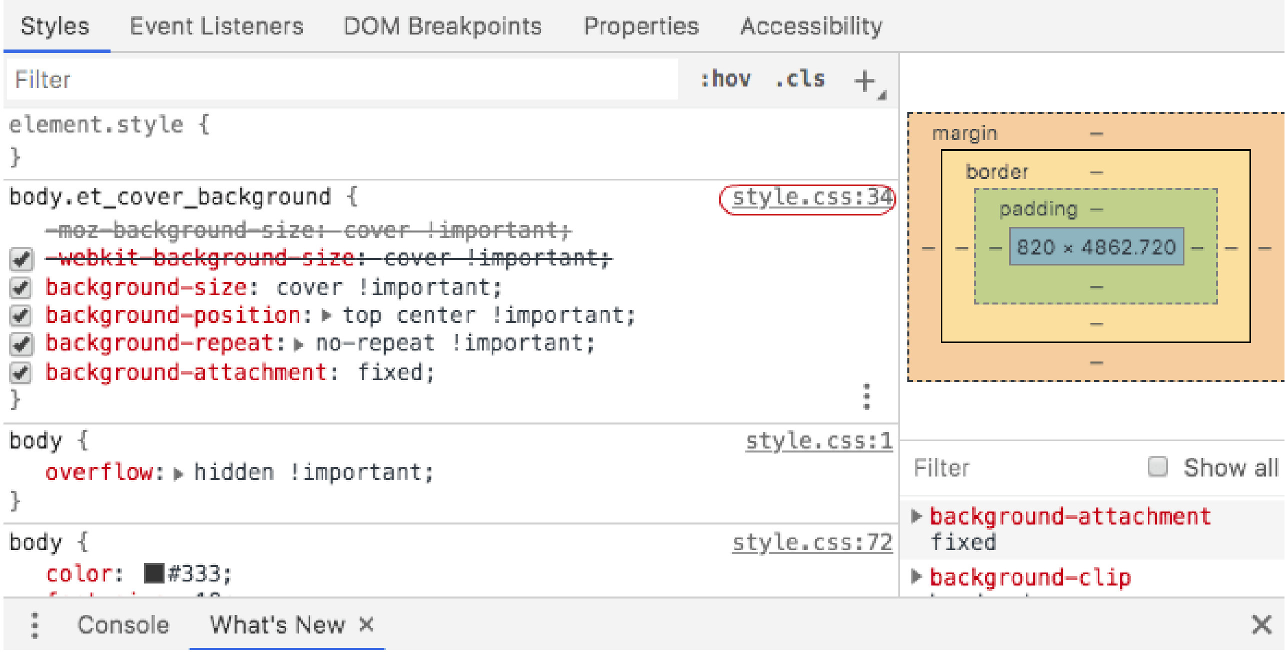
Finding your own documents is already painful. It’s even worse for your teams.

Probably going to leave that black.

3. See what they built the website with

When you visit a website, it’s not always easy to see what CMS was used to build it. But you can find out easily with Developer Tools.

Open the site and Developer Tools, then look at the top right of the Styles panel in the Elements tab:



```
element.style {
}
body.et_cover_background {                              style.css:34
  ~~-moz-background-size: cover !important;~~
☑ ~~-webkit-background-size: cover !important;~~
☑ background-size: cover !important;
☑ background-position: ▶ top center !important;
☑ background-repeat: ▶ no-repeat !important;
☑ background-attachment: fixed;
}
body {                                                  style.css:1
  overflow: ▶ hidden !important;
}
body {                                                  style.css:72
  color: ■#333;
```

```
▶ background-attachment
  fixed
▶ background-clip
```

Right-click or Ctrl-click on the link that starts “style.css” and select “Copy link address.” Paste the link into a new tab and it will tell you the CDN and theme used to build the site—in this case, it’s built using Divi on WordPress.

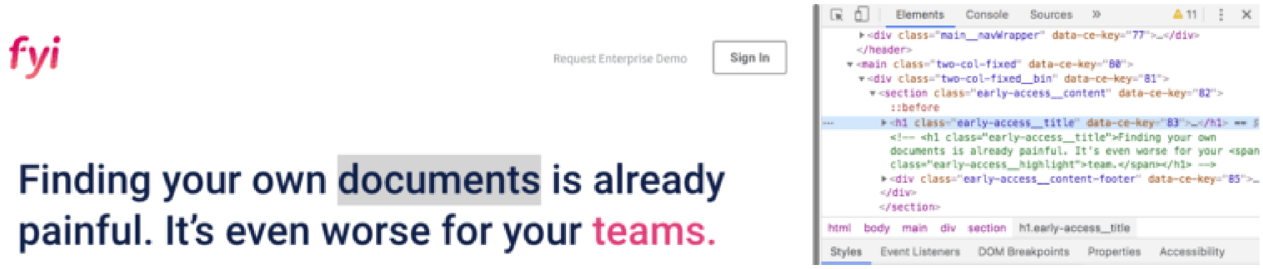
4. Write into the website

If you’re working up copy for a new web page, you can go to one that already exists and write directly into it with Developer Tools. So if layout or sizing is an issue, you can check your copy still looks good in the environment it’s going to be displayed.

Here’s how.

Open Developer Tools on a web page—you can do this with any web page.

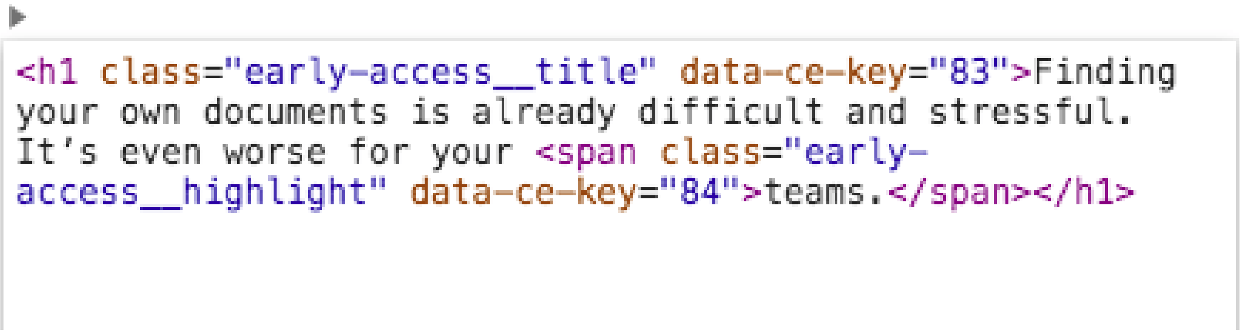
With the Elements tab open, right-click a piece of text on the page. You’ll see the relevant HTML highlighted:



```
    ▶<div class="main__navWrapper" data-ce-key="77">…</div>
    </header>
  ▼<main class="two-col-fixed" data-ce-key="80">
    ▼<div class="two-col-fixed__bin" data-ce-key="81">
      ▼<section class="early-access__content" data-ce-key="82">
          ::before
...     ▶<h1 class="early-access__title" data-ce-key="83">…</h1> == $
          <!-- <h1 class="early-access__title">Finding your own
          documents is already painful. It's even worse for your <span
          class="early-access__highlight">team.</span></h1> -->
        ▶<div class="early-access__content-footer" data-ce-key="85">…
        </div>
        </section>
```

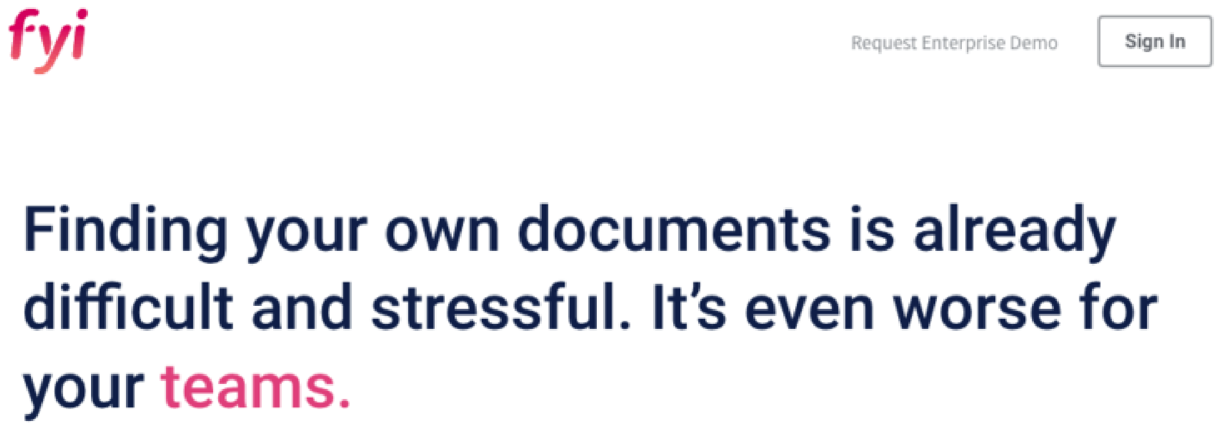
Find the three vertical dots on the left side of the HTML for that element, click them, and select "Edit as HTML."

Even if you don't know HTML from the DMV, you can still do this: all you're looking for is the text you clicked on, inside all the code. The HTML is all color-coded, while the text is black. Click on it, and you can type in whatever you like:



```
<h1 class="early-access__title" data-ce-key="83">Finding
your own documents is already difficult and stressful.
It's even worse for your <span class="early-
access__highlight" data-ce-key="84">teams.</span></h1>
```

Click on the background to exit the HTML and your changes appear in the page itself:



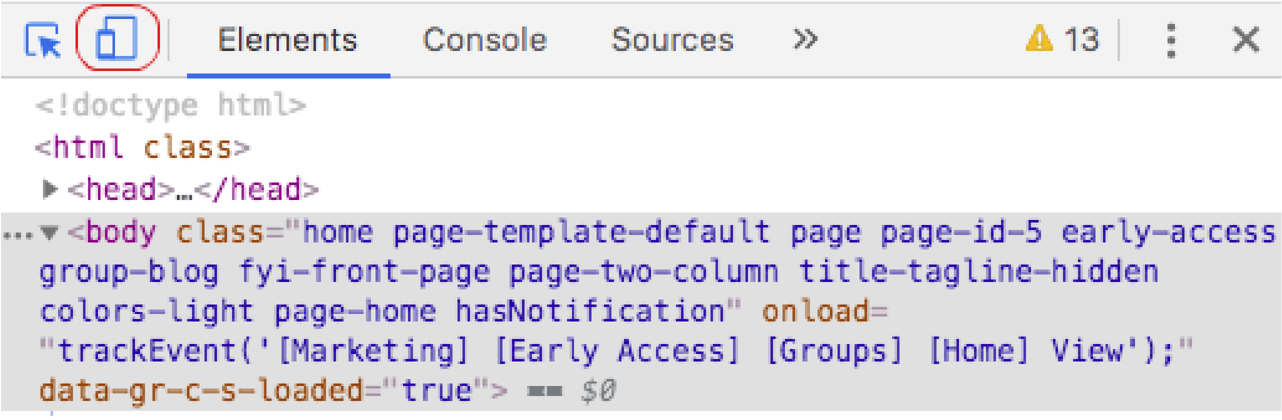
Finding your own documents is already difficult and stressful. It's even worse for your teams.

Yes, it looked better before. But this way you can find out for sure, and tailor your copy precisely to the layout and design it will be used with.

5. See it on mobile

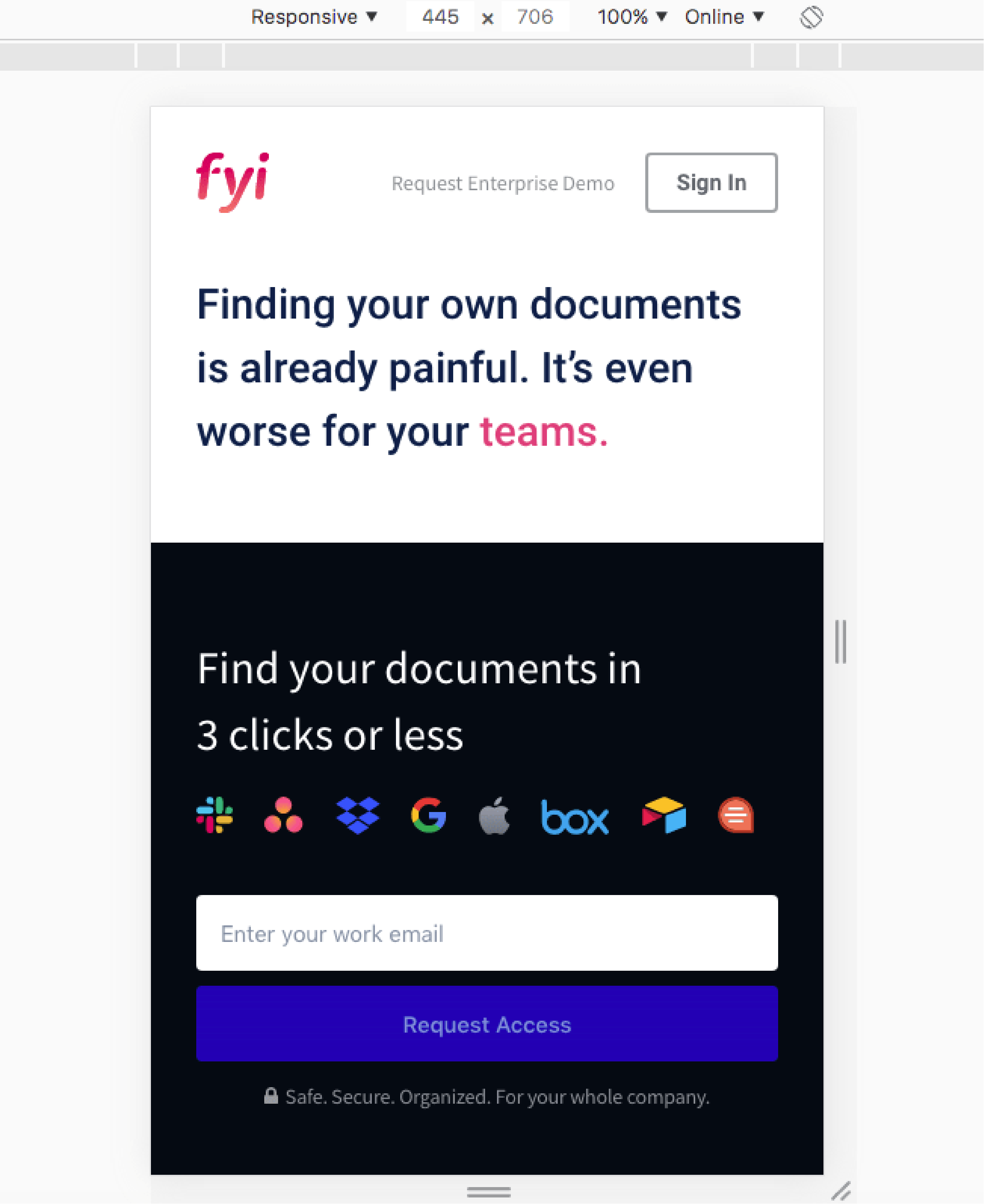
...and on any other viewport you want. Developer Tools lets you manage viewport appearance using both preselected sizes and free scaling.

Open Developer Tools and look at the top menu bar, where Elements, Console, and Source are displayed. You'll see two icons on the far left of that bar; the one that looks like a stack of mobile devices is what we're after.



```
<!doctype html>
<html class>
 ▶<head>…</head>
…▼<body class="home page-template-default page page-id-5 early-access
  group-blog fyi-front-page page-two-column title-tagline-hidden
  colors-light page-home hasNotification" onload=
  "trackEvent('[Marketing] [Early Access] [Groups] [Home] View');"
  data-gr-c-s-loaded="true"> == $0
```

Click that and you'll see the site you're on displayed as if you were a mobile visitor:



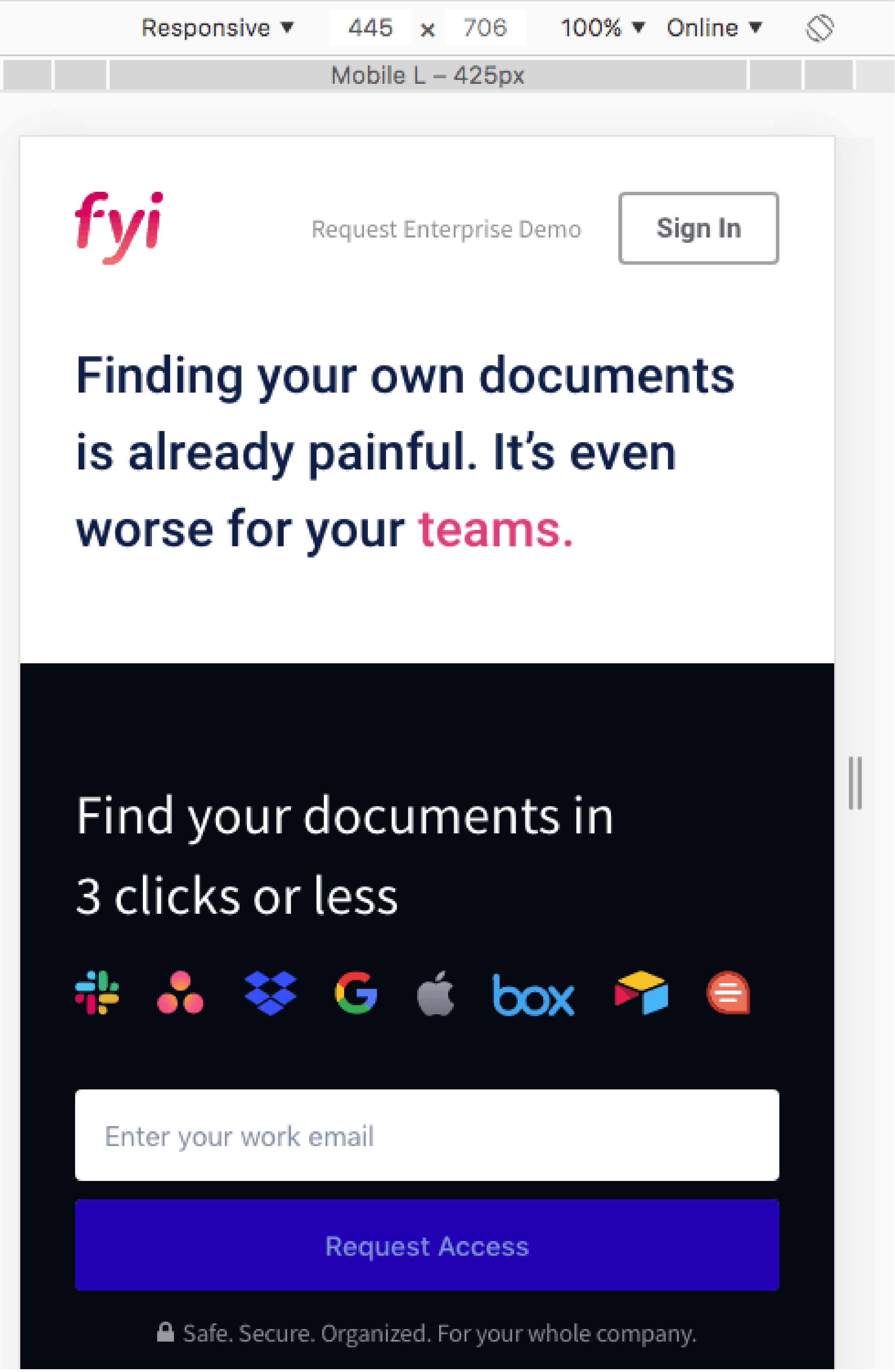
Finding your own documents is already painful. It's even worse for your teams.

Find your documents in 3 clicks or less

Enter your work email

Safe. Secure. Organized. For your whole company.

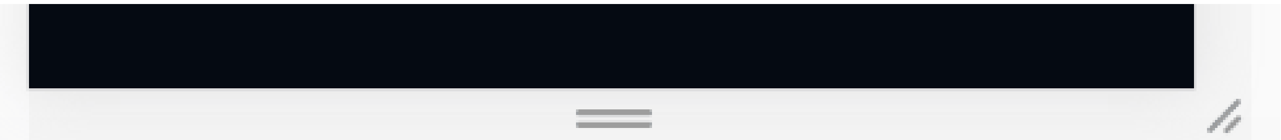
That's cool, but there's more you can do with these controls. The icon lets you toggle between mobile and desktop versions of the site. And the controls at the top of the screen let you adjust viewport size. Here's how it looks on a mobile with a big screen:



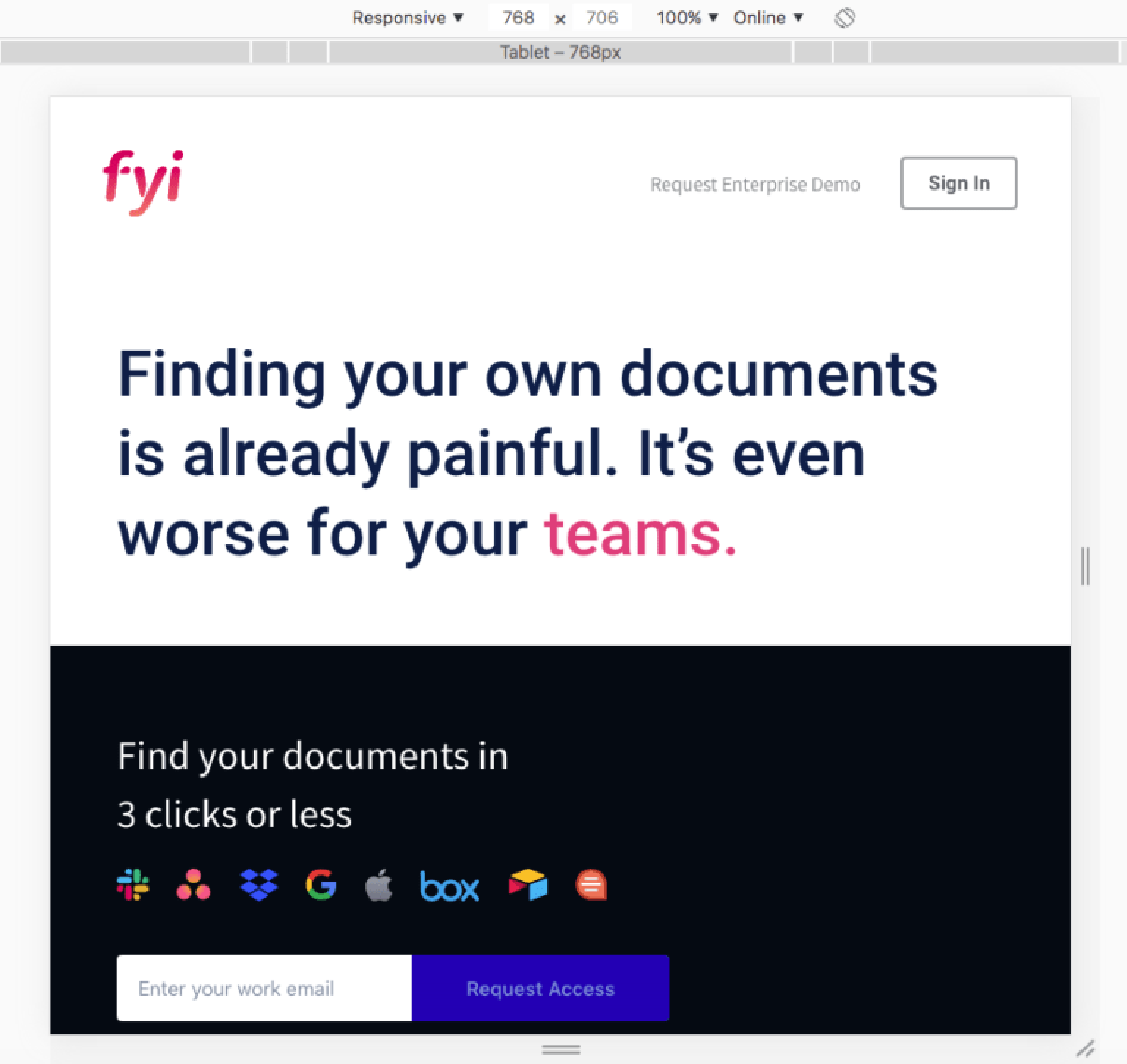
Finding your own documents is already painful. It’s even worse for your teams.

Find your documents in 3 clicks or less

Safe. Secure. Organized. For your whole company.



Here's how it looks on a tablet:



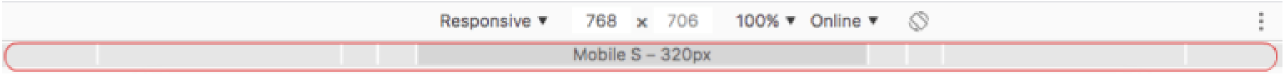
Responsive ▾ 768 × 706 100% ▾ Online ▾

Tablet – 768px

Finding your own documents is already painful. It's even worse for your teams.

Find your documents in
3 clicks or less

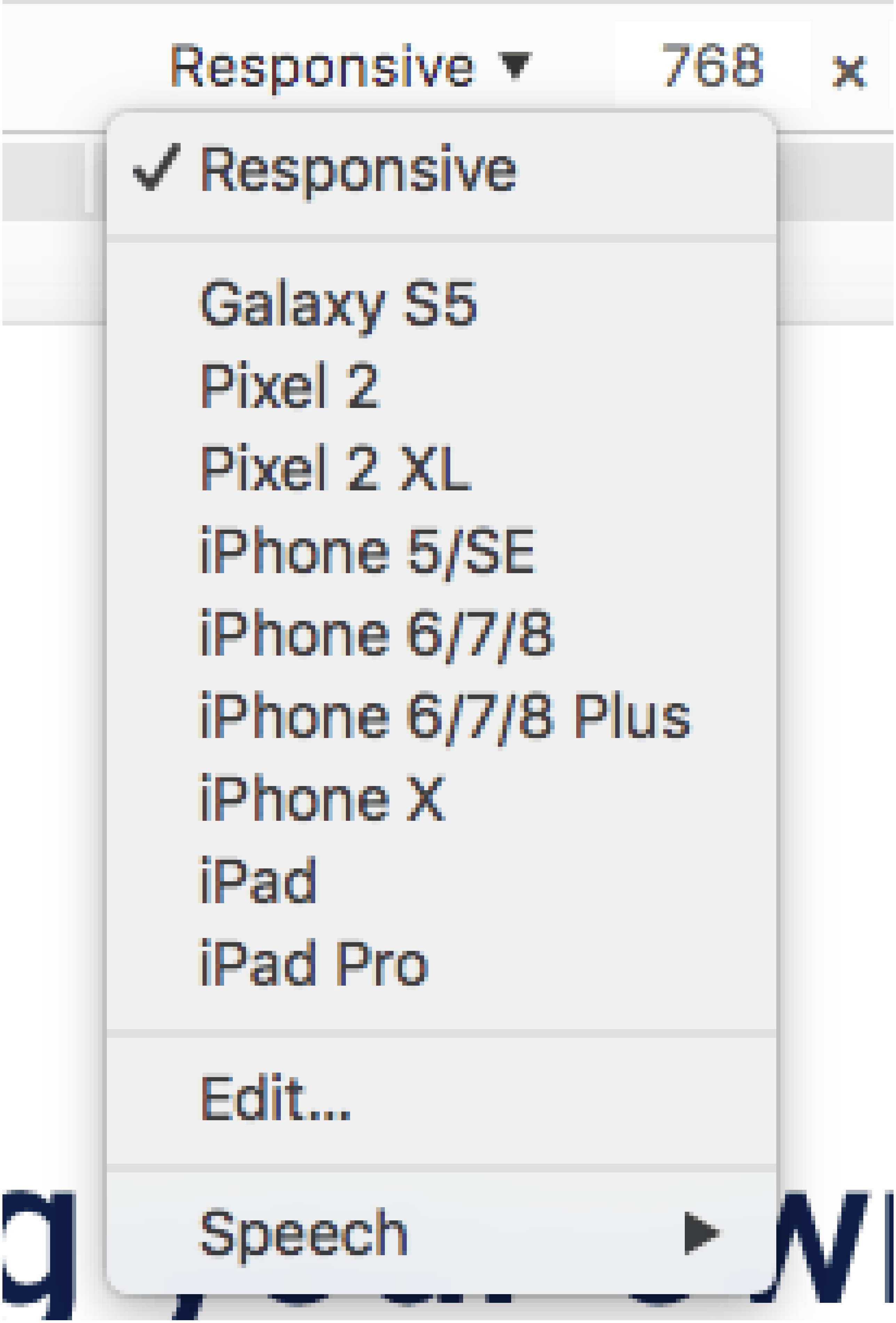
Use these controls to choose the viewport size you want:



Responsive ▾ 768 × 706 100% ▾ Online ▾

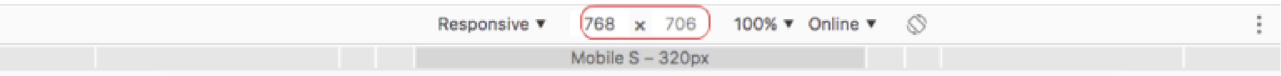
Mobile S – 320px

Use this dropdown to see what the site looks like on different devices:

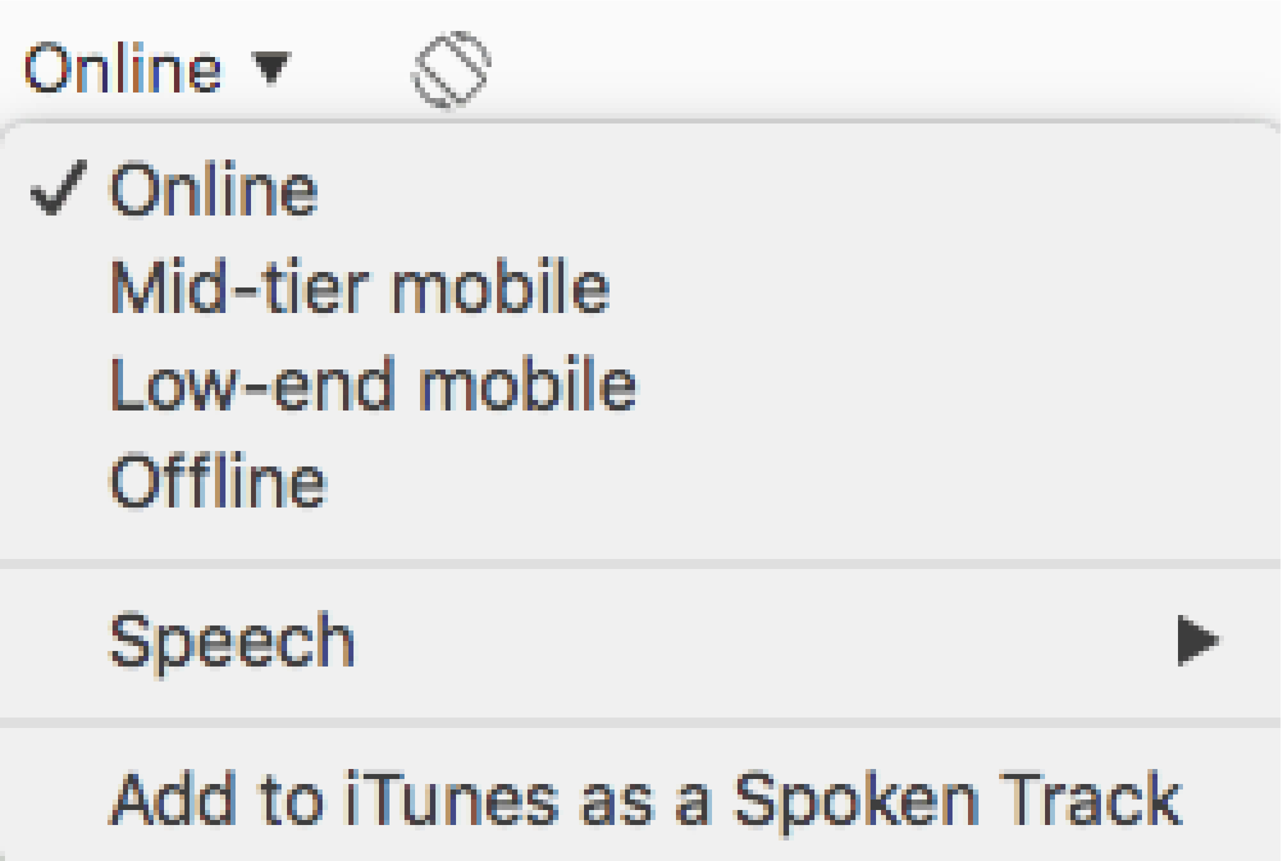


Responsive ▼
768
×
✓ Responsive
Galaxy S5
Pixel 2
Pixel 2 XL
iPhone 5/SE
iPhone 6/7/8
iPhone 6/7/8 Plus
iPhone X
iPad
iPad Pro
Edit...
Speech ▶

Set viewport size yourself:



And even mimic different connection qualities as you navigate the site:



You can do all that and more from the menu bar at the top.

What about if you want to stretch the uses for Developer Tools a little more?

5 advanced uses for Chrome Developer Tools

We're not going to dive deep into the ways developers use Developer Tools—this is a beginner's guide. But here are some things you can use Developer Tools for when you're feeling a little more confident.

To set your preferences for Developer Tools, manage them through Chrome Settings. Open the Chrome menu, then go to Settings, or type **chrome://settings/** into your omnibar.

If you know your way around Developer Tools and these tips still feel pretty basic, there’s a whole new set of in-testing tools available through Chrome Flags. Type **chrome://flags#enable-devtools-experiments** into your omnibar and you’ll be able to customize Developer Tools.

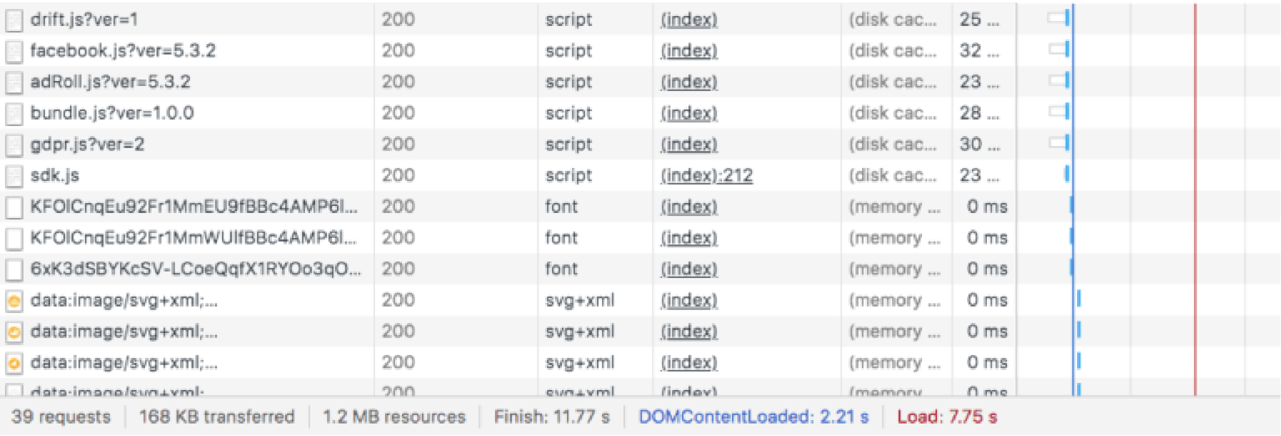
1. Identify load speed issues

Website load speed is a massive deal. Slow websites kill conversions and visitor numbers, and because Google knows this, it penalizes slow-loading sites. So if your site takes 14 seconds to load, that’s SEO poison too.

Developer Tools will tell you a lot about how and why your site loads slow.

Open your site, then open Developer Tools and go to Network. If you see a “Perform a request” message, reload the page.

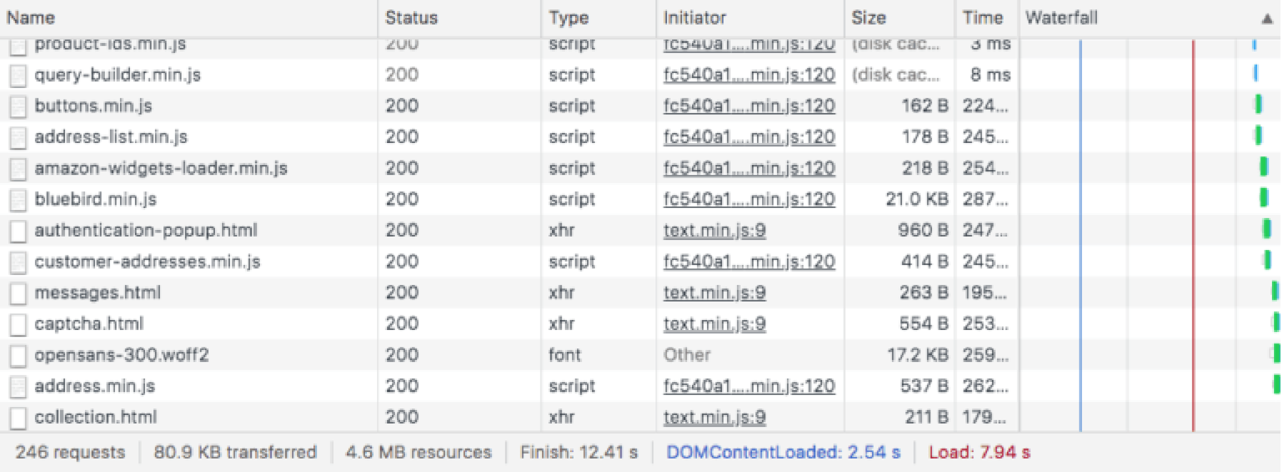
Now you can look down the list of events and resources and see what’s taking the time to load.



drift.js?ver=1	200	script	(index)	(disk cac...	25 ...
facebook.js?ver=5.3.2	200	script	(index)	(disk cac...	32 ...
adRoll.js?ver=5.3.2	200	script	(index)	(disk cac...	23 ...
bundle.js?ver=1.0.0	200	script	(index)	(disk cac...	28 ...
gdpr.js?ver=2	200	script	(index)	(disk cac...	30 ...
sdk.js	200	script	(index):212	(disk cac...	23 ...
KFOlCnqEu92Fr1MmEU9fBBc4AMP6l...	200	font	(index)	(memory ...	0 ms
KFOlCnqEu92Fr1MmWUlfBBc4AMP6l...	200	font	(index)	(memory ...	0 ms
6xK3dSBYKcSV-LCoeQqfX1RYOo3qO...	200	font	(index)	(memory ...	0 ms
data:image/svg+xml;...	200	svg+xml	(index)	(memory ...	0 ms
data:image/svg+xml;...	200	svg+xml	(index)	(memory ...	0 ms
data:image/svg+xml;...	200	svg+xml	(index)	(memory ...	0 ms
data:image/svg+xml;	200	svg+xml	(index)	(memory	0 ms

39 requests | 168 KB transferred | 1.2 MB resources | Finish: 11.77 s | DOMContentLoaded: 2.21 s | Load: 7.75 s

In this instance, there are 39 requests and just 1.2 MB in resources, which is small. Compare that to this site:



Name	Status	Type	Initiator	Size	Time	Waterfall
product-ids.min.js	200	script	fc540a1....min.js:120	(disk cac...	3 ms	
query-builder.min.js	200	script	fc540a1....min.js:120	(disk cac...	8 ms	
buttons.min.js	200	script	fc540a1....min.js:120	162 B	224...	
address-list.min.js	200	script	fc540a1....min.js:120	178 B	245...	
amazon-widgets-loader.min.js	200	script	fc540a1....min.js:120	218 B	254...	
bluebird.min.js	200	script	fc540a1....min.js:120	21.0 KB	287...	
authentication-popup.html	200	xhr	text.min.js:9	960 B	247...	
customer-addresses.min.js	200	script	fc540a1....min.js:120	414 B	245...	
messages.html	200	xhr	text.min.js:9	263 B	195...	
captcha.html	200	xhr	text.min.js:9	554 B	253...	
opensans-300.woff2	200	font	Other	17.2 KB	259...	
address.min.js	200	script	fc540a1....min.js:120	537 B	262...	
collection.html	200	xhr	text.min.js:9	211 B	179...	

246 requests | 80.9 KB transferred | 4.6 MB resources | Finish: 12.41 s | DOMContentLoaded: 2.54 s | Load: 7.94 s

246 requests, 4.6 MB in resources. Big difference, but what difference does it make to users?

The answer to that is on the bottom of the panel. Look for the last three numbers: Finish, DOMContentLoaded, and Load.

DOMContentLoad tells you how long it took to load the HTML for the page before anything else—CSS, JavaScript, whatever else is on there. There’s just 0.29 seconds’ difference between these two sites in DOMContentLoad time. Don’t look for huge differences here.

Load tells us when the page is fully loaded; Finish tells us when all the elements and code in the page is done including asynchronous stuff. That’s a lot of what’s moving around in the second, slower page that’s shown here. You can glance at the table and see a ton of scripts for Amazon and other third-party sites running.

What you can also do is check the usual suspects: inline CSS, images, lots of requests, and a bunch of JS and Flash. It all shows up in the waterfall in the top.

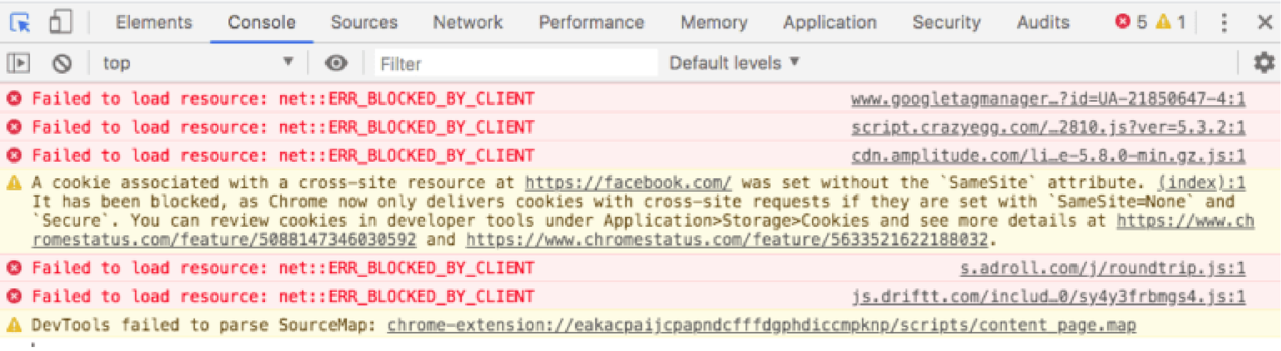
2. Catch non-working plugins

Nearly every website relies on some kind of third-party plugin. Whether it’s tracking code from HotJar or Crazy Egg, form or popup tools, or chat addons like Intercom or Zendesk Chat, it’s crucial to the business the website is meant to perform. When it goes wrong it’s not always easy to know it’s happening, never mind how or why.

There's an easy way to catch it in Developer Tools.

Open the site you want to check, open Developer Tools and go to the Console tab.

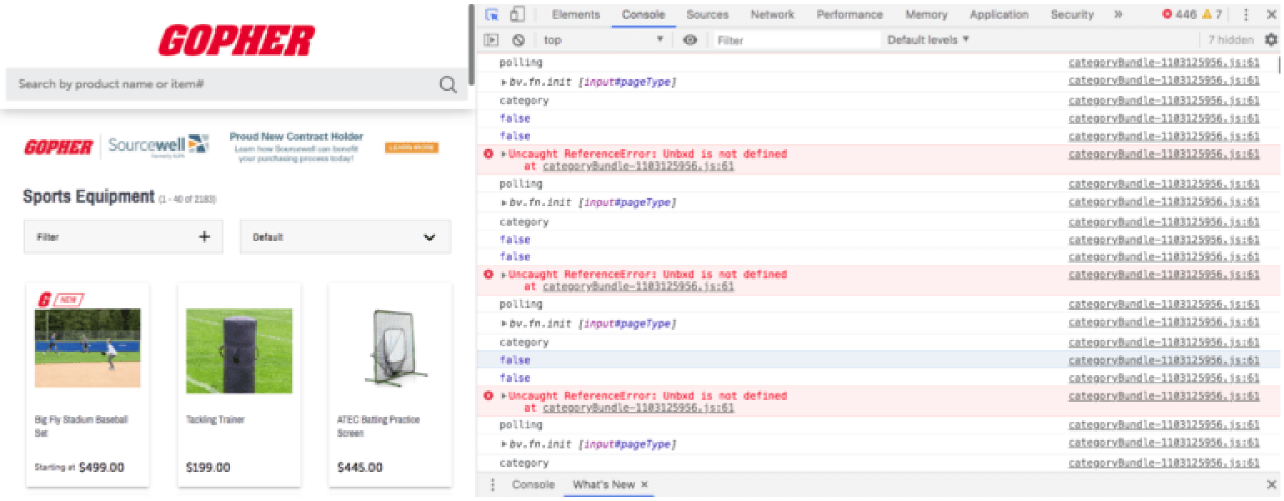
Errors will be flagged in red:



```
Failed to load resource: net::ERR_BLOCKED_BY_CLIENT          www.googletagmanager…?id=UA-21850647-4:1
Failed to load resource: net::ERR_BLOCKED_BY_CLIENT          script.crazyegg.com/…2810.js?ver=5.3.2:1
Failed to load resource: net::ERR_BLOCKED_BY_CLIENT          cdn.amplitude.com/li…e-5.8.0-min.gz.js:1
A cookie associated with a cross-site resource at https://facebook.com/ was set without the `SameSite` attribute. (index):1
It has been blocked, as Chrome now only delivers cookies with cross-site requests if they are set with `SameSite=None` and
`Secure`. You can review cookies in developer tools under Application>Storage>Cookies and see more details at https://www.ch
romestatus.com/feature/5088147346030592 and https://www.chromestatus.com/feature/5633521622188032.
Failed to load resource: net::ERR_BLOCKED_BY_CLIENT                   s.adroll.com/j/roundtrip.js:1
Failed to load resource: net::ERR_BLOCKED_BY_CLIENT          js.driftt.com/includ…0/sy4y3frbmgs4.js:1
DevTools failed to parse SourceMap: chrome-extension://eakacpaijcpapndcffdgphdiccmpknp/scripts/content_page.map
```

If you're seeing a page full of "BLOCKED_BY_CLIENT" it's probably because you have an adblocker running.

On the other hand, this company hasn't put the code for their tracking tool into their website correctly:



```
polling                                                  categoryBundle-1103125956.js:61
▸ bv.fn.init [input#pageType]                            categoryBundle-1103125956.js:61
category                                                 categoryBundle-1103125956.js:61
false                                                    categoryBundle-1103125956.js:61
false                                                    categoryBundle-1103125956.js:61
▸ Uncaught ReferenceError: Unbxd is not defined          categoryBundle-1103125956.js:61
    at categoryBundle-1103125956.js:61
polling                                                  categoryBundle-1103125956.js:61
▸ bv.fn.init [input#pageType]                            categoryBundle-1103125956.js:61
category                                                 categoryBundle-1103125956.js:61
false                                                    categoryBundle-1103125956.js:61
false                                                    categoryBundle-1103125956.js:61
▸ Uncaught ReferenceError: Unbxd is not defined          categoryBundle-1103125956.js:61
    at categoryBundle-1103125956.js:61
polling                                                  categoryBundle-1103125956.js:61
▸ bv.fn.init [input#pageType]                            categoryBundle-1103125956.js:61
category                                                 categoryBundle-1103125956.js:61
false                                                    categoryBundle-1103125956.js:61
false                                                    categoryBundle-1103125956.js:61
▸ Uncaught ReferenceError: Unbxd is not defined          categoryBundle-1103125956.js:61
    at categoryBundle-1103125956.js:61
polling                                                  categoryBundle-1103125956.js:61
▸ bv.fn.init [input#pageType]                            categoryBundle-1103125956.js:61
category                                                 categoryBundle-1103125956.js:61
```

Those "Uncaught Reference" errors are the UNBXD tracking code not working. The reason is because "there is a non-existent variable referenced somewhere. This variable needs to be declared, or you need to make sure it is available in your current script or scope," according to Mozilla's JavaScript tutorial.

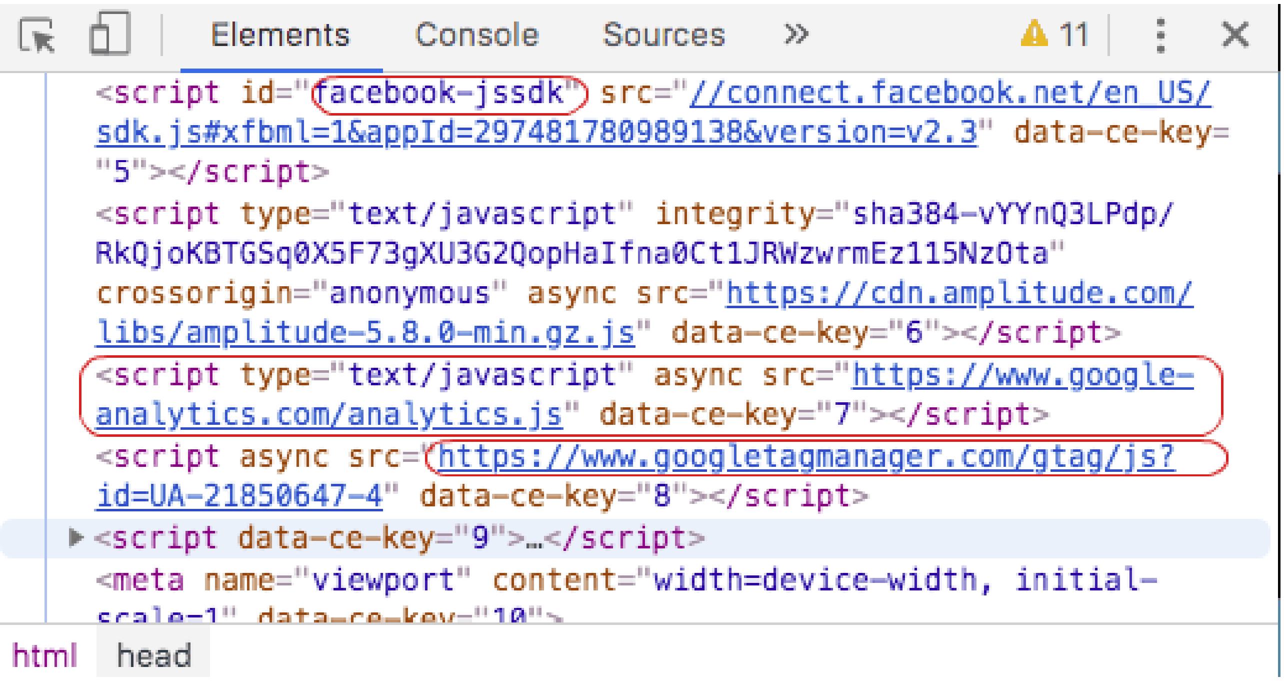
Even if you have no clue what that means, you do know that the tool isn't working. That's vital information that your development team, marketing department, and a bunch of others at your company will want to know.

3. See what technology a site is using

You're selling tech, but you don't know if this business is a potential customer. Some technographic information would be nice. Let's open Developer Tools and get some.

With Developer Tools open in the Elements tab, find the <head> tag in the HTML and click on it to expand the section.

You'll see all the tracking JavaScript for the site:



```
<script id="facebook-jssdk" src="//connect.facebook.net/en_US/
sdk.js#xfbml=1&appId=297481780989138&version=v2.3" data-ce-key=
"5"></script>
<script type="text/javascript" integrity="sha384-vYYnQ3LPdp/
RkQjoKBTGSq0X5F73gXU3G2QopHaIfna0Ct1JRWzwrmEz115NzOta"
crossorigin="anonymous" async src="https://cdn.amplitude.com/
libs/amplitude-5.8.0-min.gz.js" data-ce-key="6"></script>
<script type="text/javascript" async src="https://www.google-
analytics.com/analytics.js" data-ce-key="7"></script>
<script async src="https://www.googletagmanager.com/gtag/js?
id=UA-21850647-4" data-ce-key="8"></script>
▶<script data-ce-key="9">…</script>
<meta name="viewport" content="width=device-width, initial-
scale=1" data-ce-key="10">
```

Facebook, Tag Manager, Google Analytics… if they're using anything more specialized that will show up here too.

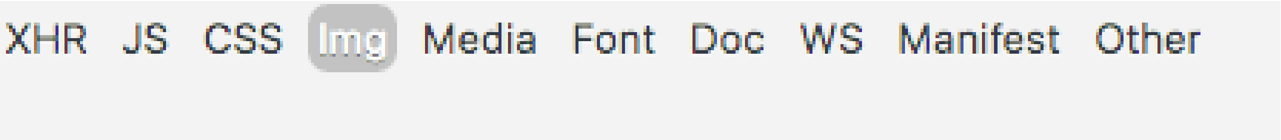
4. Logo Grab

You're building a deck or some other piece of content and you need logos, but you can't find the website's press kit. What to do?

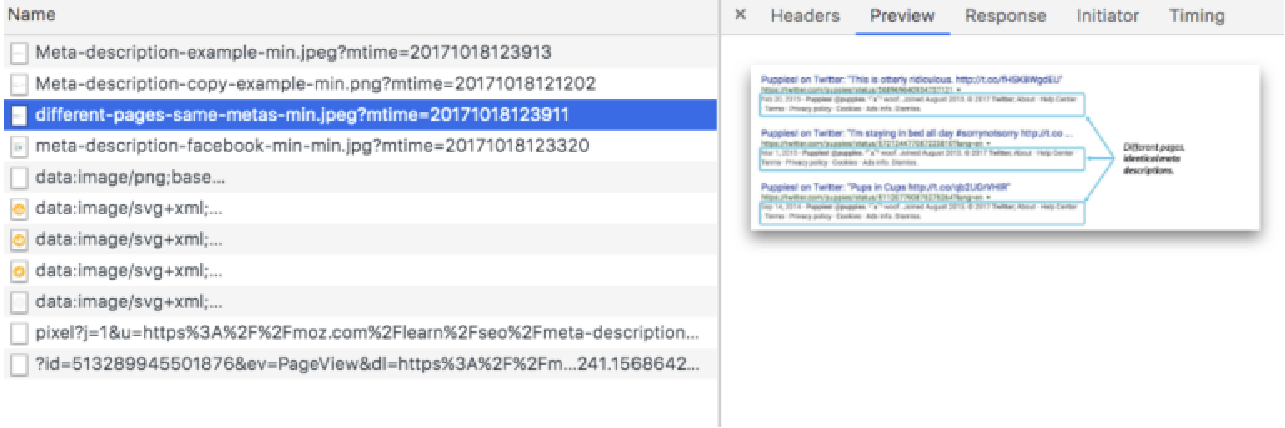
Mouse over the logo on the homepage, right-click or Ctrl-click and select "Inspect." Developer tools will open, and the code for the logo will be highlighted in the HTML displayed on the Elements tab. Look in the code for a full URL that starts **https://**.

Copy and paste that into a new browser tab and you have the logo, usually as a fairly large PNG.

The alternative is to open Developer Tools and head to the Network tab, then look for the image you want. You don't have to scour all the elements on the page manually. You can filter by type for images in this panel:



When you find the one you're looking for, it's displayed in the Preview panel alongside.


×
Headers
Preview
Response
Initiator
Timing
Puppies! on Twitter: "This is otterly ridiculous. http://t.co/fHSK8WgdEU"
Puppies! on Twitter: "I'm staying in bed all day #sorrynotsorry http://t.co ...
Puppies! on Twitter: "Pups in Cups http://t.co/gb2UG/VHIR"
Different pages, identical meta descriptions.

Name
Meta-description-example-min.jpeg?mtime=20171018123913
Meta-description-copy-example-min.png?mtime=20171018121202
different-pages-same-metas-min.jpeg?mtime=20171018123911
meta-description-facebook-min-min.jpg?mtime=20171018123320
data:image/png;base...
data:image/svg+xml;...
data:image/svg+xml;...
data:image/svg+xml;...
data:image/svg+xml;...
pixel?j=1&u=https%3A%2F%2Fmoz.com%2Flearn%2Fseo%2Fmeta-description...
?id=513289945501876&ev=PageView&dl=https%3A%2F%2Fm...241.1568642...

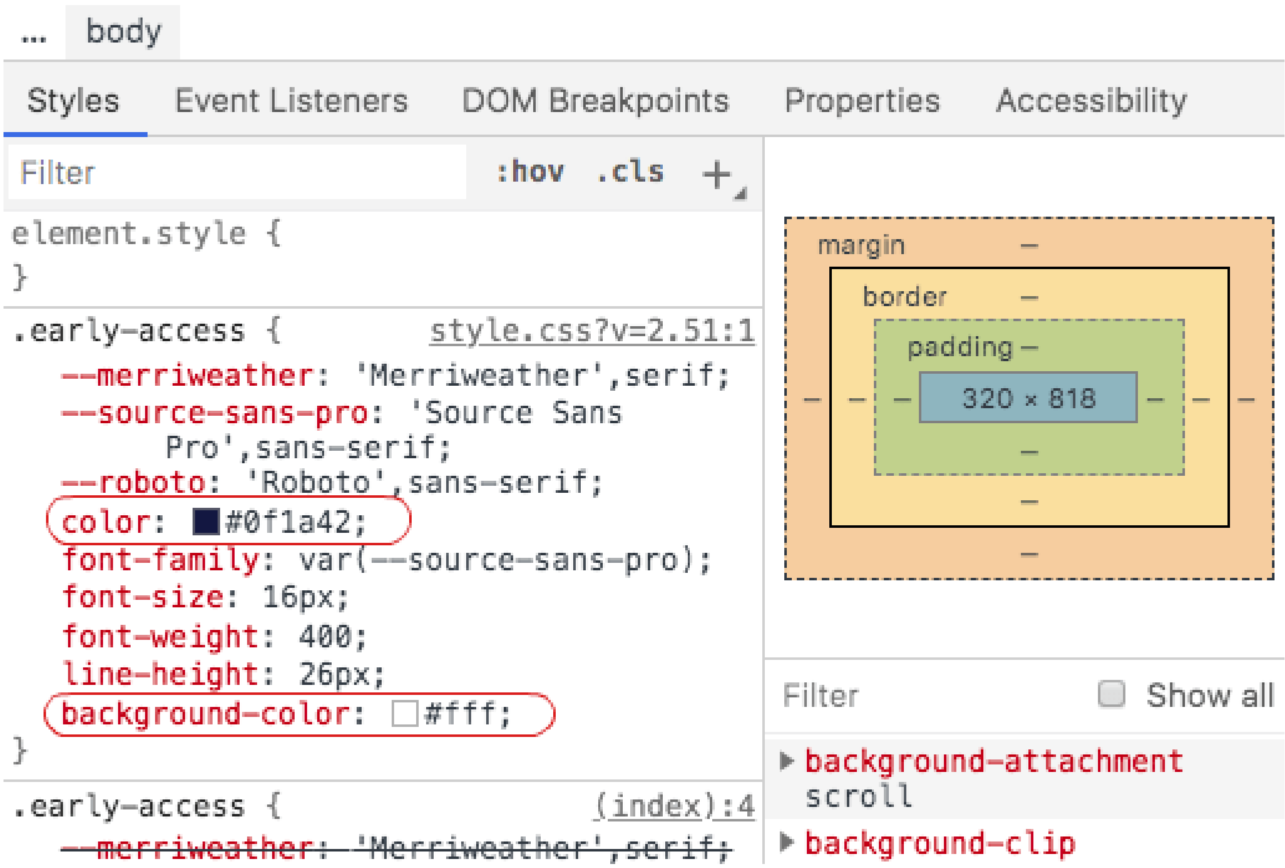
Right-click/Ctrl+click on the image preview to get the option to save the image or download it, or save it as a data URI.

The only time this doesn't work is when the logo is an SVG that's actually been drawn on the page. That's when you'll have to take a screenshot or search Google images.

5. Color Picker

Chrome has a surprisingly powerful built-in color picker. You can get hex, RGBA, or HSLA color codes for any element of a website you're looking at, from logos to text to background colors.

Open Developer Tools and stay in the Elements tab. Look at the bottom half of the tab, in the Styles tab, for a CSS element that has the quality of color:



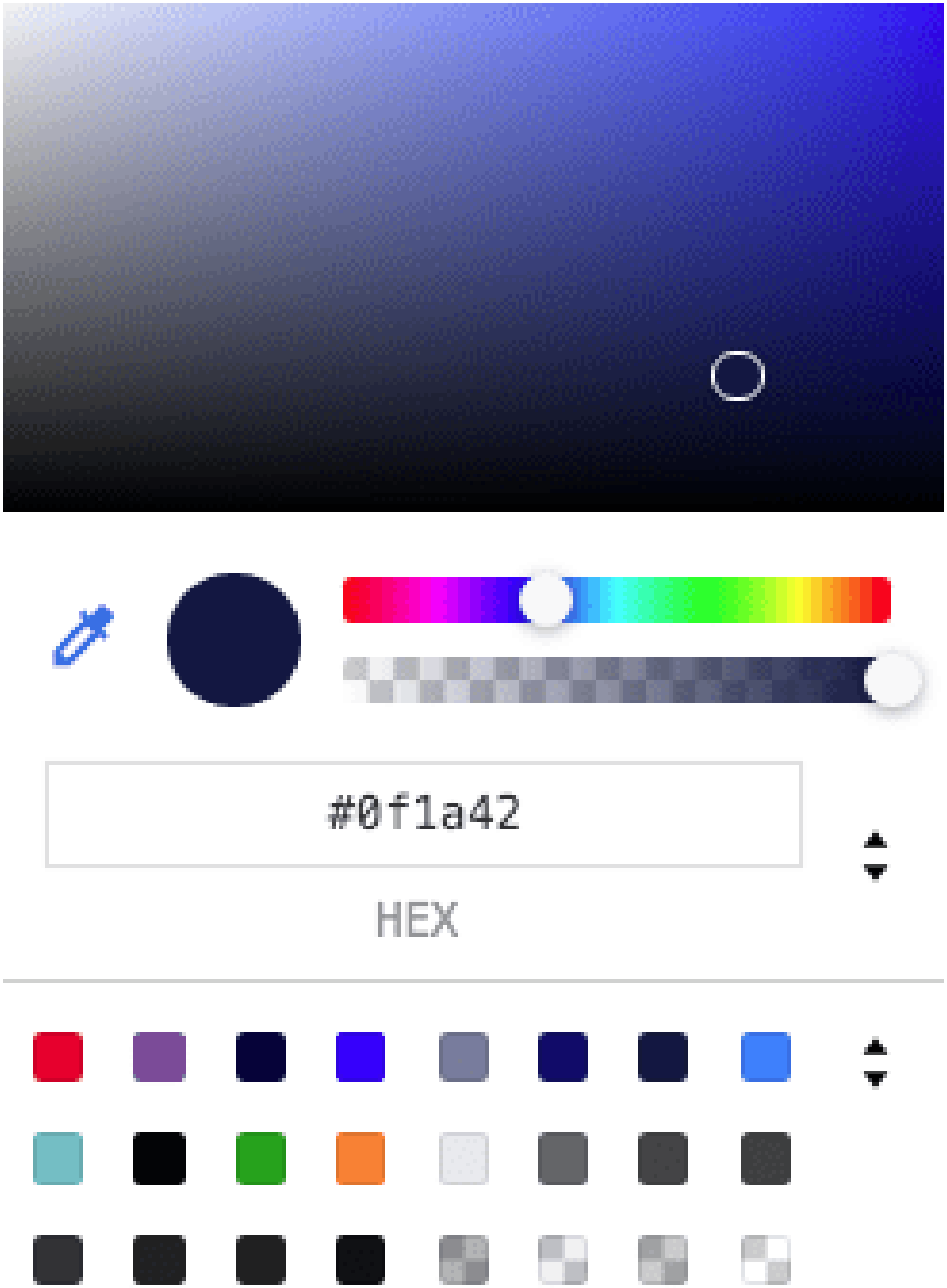
```
element.style {
}
.early-access {                style.css?v=2.51:1
   --merriweather: 'Merriweather',serif;
   --source-sans-pro: 'Source Sans
         Pro',sans-serif;
   --roboto: 'Roboto',sans-serif;
   color: ■#0f1a42;
   font-family: var(--source-sans-pro);
   font-size: 16px;
   font-weight: 400;
   line-height: 26px;
   background-color: □#fff;
}
.early-access {                      (index):4
   --merriweather: 'Merriweather',serif;
```

```
▶ background-attachment
  scroll
▶ background-clip
```

So far, so good: you can see the color information for that element.

Click on the CSS element and you’ll open the full-feature Chrome color picker. You have to click on the tiny box that actually displays the color visually, not the text.

You’ll see this:



#0f1a42

Use the eyedropper to detect the color of any element on the page, then copy the hex, RGBA, or HSLA color code from underneath.

Top Posts

What is Nira?

Over 45% of employees admit to taking documents from former employers

How 51 Leading Companies Define the IT Manager Role

Protect company documents from unauthorized access

51 Biggest Document Leaks & Data Breaches of All Time

Guides and Manuals

Best Guide on How to Secure Your Documents

The Ultimate Manual to Employee Data Security

The Ultimate Manual For Data Breaches

The Ultimate Manual To Sensitive Data

Best Guide on How to Defend Your Remote Business from Cyberattacks

The Ultimate Manual To Vendor Assessment

The Ultimate Manual For Data Loss Prevention (DLP)

The Ultimate Manual For Identity and Access Management (IAM)

IT Security Policy: The Complete Guide

The Ultimate Manual For Data Governance

Search

Try "Data Security"