

Web Page Debuggers for Non-Developers

Browsers have built-in tools for web developers called "Developer Tools".

This evening I'll present how non-developers can make use of these tools too via live demos.

Live Demos!

- Measuring load performance of web pages
- Checking web pages for errors and other issues
- Finding job listings with Developer Tools

Measuring Load Performance of Web Pages

If you're a web developer, you know why you want to measure load page times. But what about the rest of us?

How many here are annoyed with how long web pages load?

I'm guessing one or two of you.

You can do things without changing web pages themselves to make them faster. And it's good to be able to quantify exactly how much those changes have helped.

For example, changing the configuration of your browser can improve performance -- such as using an ad blocker.

For my examples, I'll use Firefox. I use Firefox 99% of the time, so I know it well. However, everything I do here can also be done in Chrome as well.

To measure performance, I'll use the feature of Firefox that's part of Developer Tools called Network Interface. There's two ways to bring it up. You can go through the hamburger menu with "Web Developer > Network", or you can use the hotkey Control-Shift-E.

We need a web page that is ad-heavy. Most newspaper sites fit the bill, so I'll pick on The Washington Post.

First, I'll open a new tab and activate the Network Interface. Now I'll load the Washington Post front page. Down at the bottom in the "load:" field, you can see the page loaded in X.XXX seconds.

My favorite ad blocker is "uBlock Origin". I'll be using that for the demo. I'll activate it and reload the page.

Now when you do this test, since results can vary, I recommend reloading the page several times both with and without the ad blocker, but to save time, I'll skip doing that here.

When I reload the page after enabling uBlock Origin, you can see in the Network Interface console it is blocking a lot of URLs. This not only saves time rendering the page, but saves network bandwidth as well, so a double win.

As you can see, ads made the page load time 2X-3X longer or even more than it otherwise would.

Now onto checking web pages for errors and other problems.

Checking Web Pages for Errors and Other Issues

Admittedly, most people here don't create web pages. Those of us that don't may still find this useful.

For example, if I'm thinking of opening a new account with a bank and will be using their web interface often, I'd like to have an idea of how skilled and state-of-the-art their web developers are.

Using results from Google's web developer tool "Lighthouse", I can get me a high level overview of a web pages' quality.

For this demo, I looked at several banks including Capital One, CitiBank, and Chase. For this demo, I'll use Chase because of the problems I found.

I'll load the site's page and then click on the Lighthouse feature to generate a report. Before I run it though, I'll make sure that configured for "Desktop".

Running Lighthouse takes about 30 seconds.

Chase's web site isn't horrible, but I noticed something when I clicked on "Best Practices". Lighthouse reports the JavaScript libraries it's using has known security vulnerabilities. This would make me uneasy using Chase as my bank -- not just for these specific vulnerabilities, but because they don't have a process that regularly updates their pages' code for security vulnerabilities. I noticed this error about 2 weeks ago when researching pages, and it's still a problem.

Now let's try Lighthouse out on our favorite web site, CUCUG!

I'll open a new tab, load CUCUG, and run Lighthouse on it.

This page displays blindingly fast! And Lighthouse agrees! It's the only web site I've seen with a 100% performance score.

If we go down to "Accessibility", under tables and lists we find Lighthouse complaining about lists not starting with a `` or ``. We use the turn down, we can see what its complaining about. Now let's check the page's source with Control+U.

You can see that the list Lighthouse was commenting about from lines 65 to 70. Sure enough, it would like a `` before line 65. I'm sure Kevin now will fix that when it can!

So even if you're not a web developer, you can help developers find mistakes and get them fixed.

Finding Job Listings with Developer Tools

The last item is finding job listings in the source code of web pages! I just found out about this yesterday. If we go to wh.gov and bring up its source with Control+U, on line 9 you'll find:

```
<!-- If you're reading this, we need your help building back better.  https://usds.gov/ -->
```

USDS is the United States Digital Service.

Are there any questions?

And now back to Mark.