



Performance Testing

Performance is one of the hottest topics and, in my experience, one of the hardest to get right. There are tools that will help developers measure and improve a site's performance.

The idea behind Lighthouse is that by running it against your site, either staging or production periodically you can get a good overview of your site's performance in the categories that lighthouse measures.

It will give you a baseline to measure against, find areas to improve on and track any regresswion that may have snuck by your testing.

Lighthouse

Lighthouse is an automated tool for improving the quality of web pages. It has audits for performance, accessibility, best practices, SEO, and progressive web apps.

You can run Lighthouse in Chrome DevTools, as an extension, from the command line, or as a Node module. You give Lighthouse a URL to audit, it runs a series of audits against the page, and then it generates a report on how well the page did. Use the failing audits as indicators on how to improve the page... think of it as performance driven development.

DevTools Audit Menu

To open DevTools hit `command+I` in Macintosh and `control+I` in Windows and Linux

Lighthouse is now part of the Chrome DevTools suite of developer productivity tools. If you look at the audit menu you will see the lighthouse logo and the configuration for the tool.

Figure 1:
DevTools
under

If you haven't already navigate to the page you want to test and click the audit menu in DevTools. For the purpose of these examples we'll leave the default settings active; these may not all be necessary for all sites but give a good idea of how will your app or site work in a mid-level mobile device under less-than-optimal conditions.

CLI

The source code for lighthouse is available on its [Github Repository](#) and, from my perspective, it's the best way to run Lighthouse without having a browser open and it gives you access to the latest and greatest

To get the CLI running you have to install [Yarn](#). You can do so with your favorite package manager (Homebrew in Mac or Chocolatey in Windows) or following the instructions in Yarn's [Installation](#) page.

Once Yarn is installed, download the Lighthouse code from Github. Easiest way is to use the Git command line tool:

```
git pull --recursive https://github.com/googlechrome/lighthouse.git
```

Once it has downloaded all the dependencies we can build the tools by running `yarn build-all` in the terminal.

Lighthouse run a series of tasks in its own instance of Chrome. The results are provided in an HTML page that you can open in your default browser automatically.

The example below will use the Lighthouse CLI we just built to test a site and display the result document in the user's default browser.

```
node lighthouse-cli/ https://web-layout-experiments.firebaseio.com --view
```

NPM Module

If you need to build lighthouse into your build or CI processes or don't want to be

bothered with another package manager, there is an [NPM Plugin](#) to handle your needs.

Install it like any other plugins. I prefer global installation in this case:

```
npm i -g lighthouse
```

Then run it against an example site. Using the site from the CLI example:

```
lighthouse-cli/ https://web-layout-experiments.firebaseio.com
```