What's New In DevTools (Chrome 68)

By Kayce Basques
Technical Writer for Chrome DevTools

New to DevTools in Chrome 68:

- Eager Evaluation. As you type expressions, the Console previews the result.
- Argument hints. As you type functions, the Console shows you the expected arguments for that function.
- Function autocompletion. After typing a function call such as `document.querySelector('p')`, the Console shows you the functions and properties that the return value supports.
- ES2017 keywords in the Console. Keywords such as `await` are now available in the Console's autocomplete UI.
- Lighthouse 3.0 in the Audits panel. Faster, more consistent audits, a new UI, and new audits.
- `BigInt` support. Try out JavaScript's new arbitrary-precision integer in the Console.
- Adding property paths to the Watch pane. Add properties from the Scope pane to the Watch pane.
- "Show timestamps" moved to Settings.

Note: Check what version of Chrome you're running at `chrome://version`. If you're running an earlier version, these features won't exist. If you're running a later version, these features may have changed. Chrome auto-updates to a new major version about every 6 weeks.

Read on, or watch the video version of the release notes, below.

Assistive Console

Chrome 68 ships with a few new Console features related to autocompletion and previewing.

Eager Evaluation

When you type an expression in the Console, the Console can now show a preview of the result of that expression below your cursor.

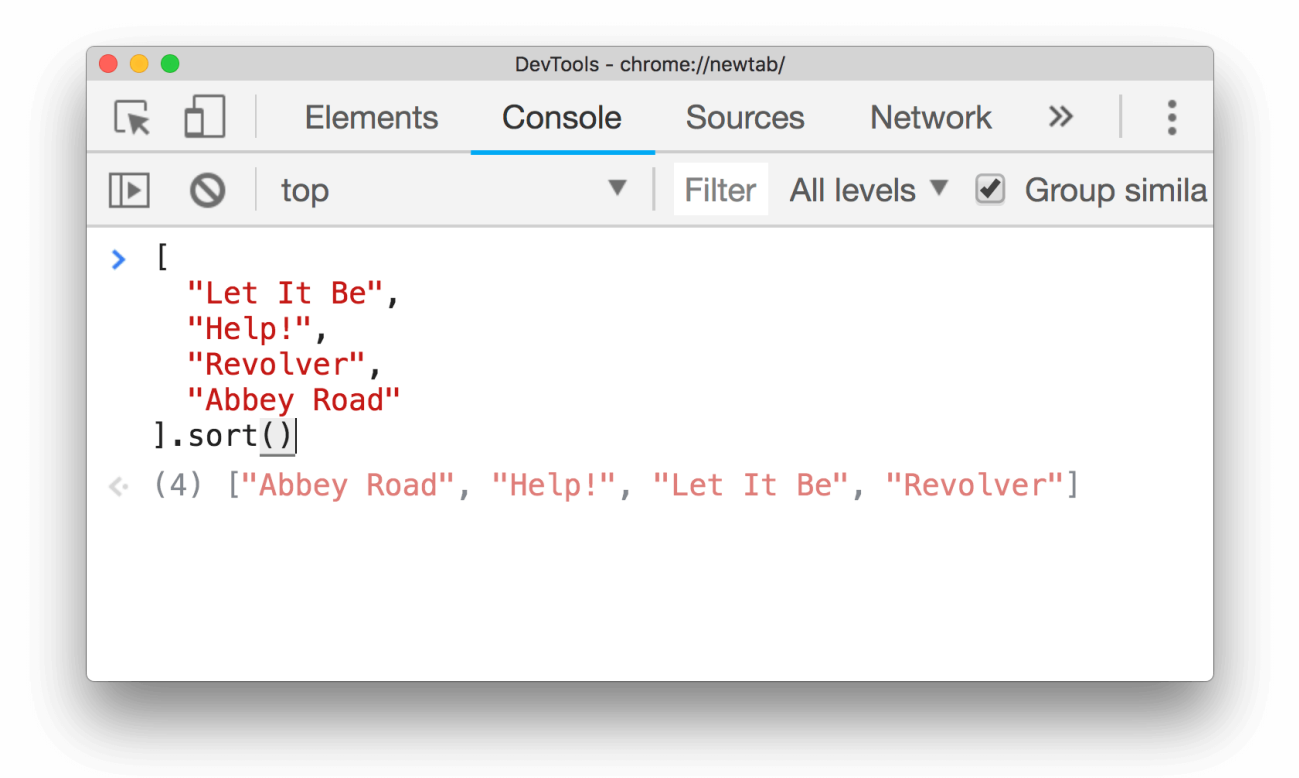

Figure 1. The Console is printing the result of the `sort()` operation before it has been explicitly executed

To enable Eager Evaluation:

1. Open the **Console**.
2. Open **Console Settings** .
3. Enable the **Eager evaluation** checkbox.

DevTools does not eager evaluate if the expression causes side effects .

Argument hints

As you're typing out functions, the Console now shows you the arguments that the function expects.

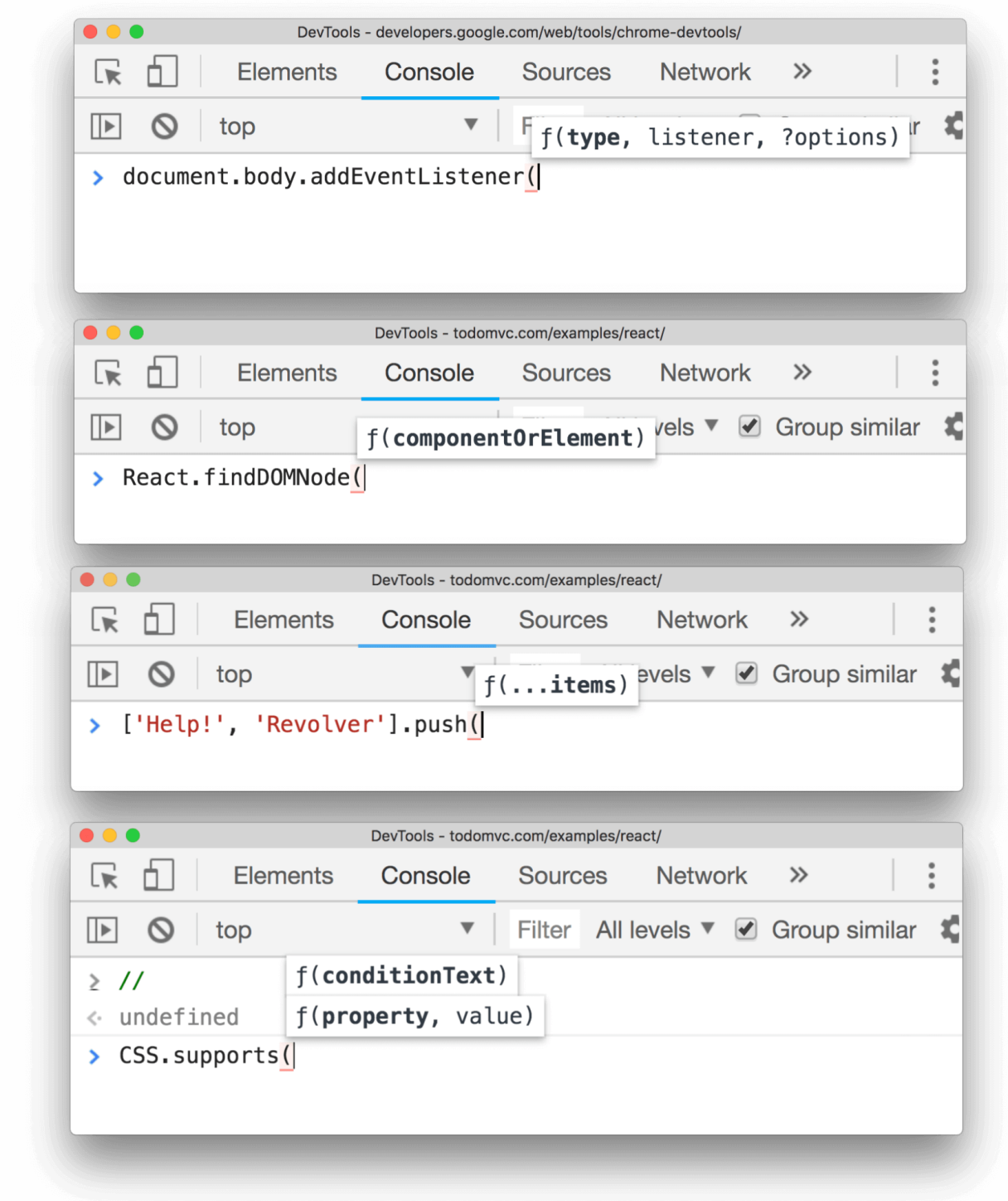

Figure 2. Various examples of argument hints in the Console

Notes:

- A question mark before an arg, such as `?options`, represents an optional arg.
- An ellipsis before an arg, such as `...items`, represents a spread.
- Some functions, such as `CSS.supports()`, accept multiple argument signatures.

Autocomplete after function executions

Note: This feature depends on Eager Evaluation, which needs to be enabled from **Console Settings** ⚙.

After enabling Eager Evaluation, the Console now also shows you which which properties and functions are available after you type out a function.

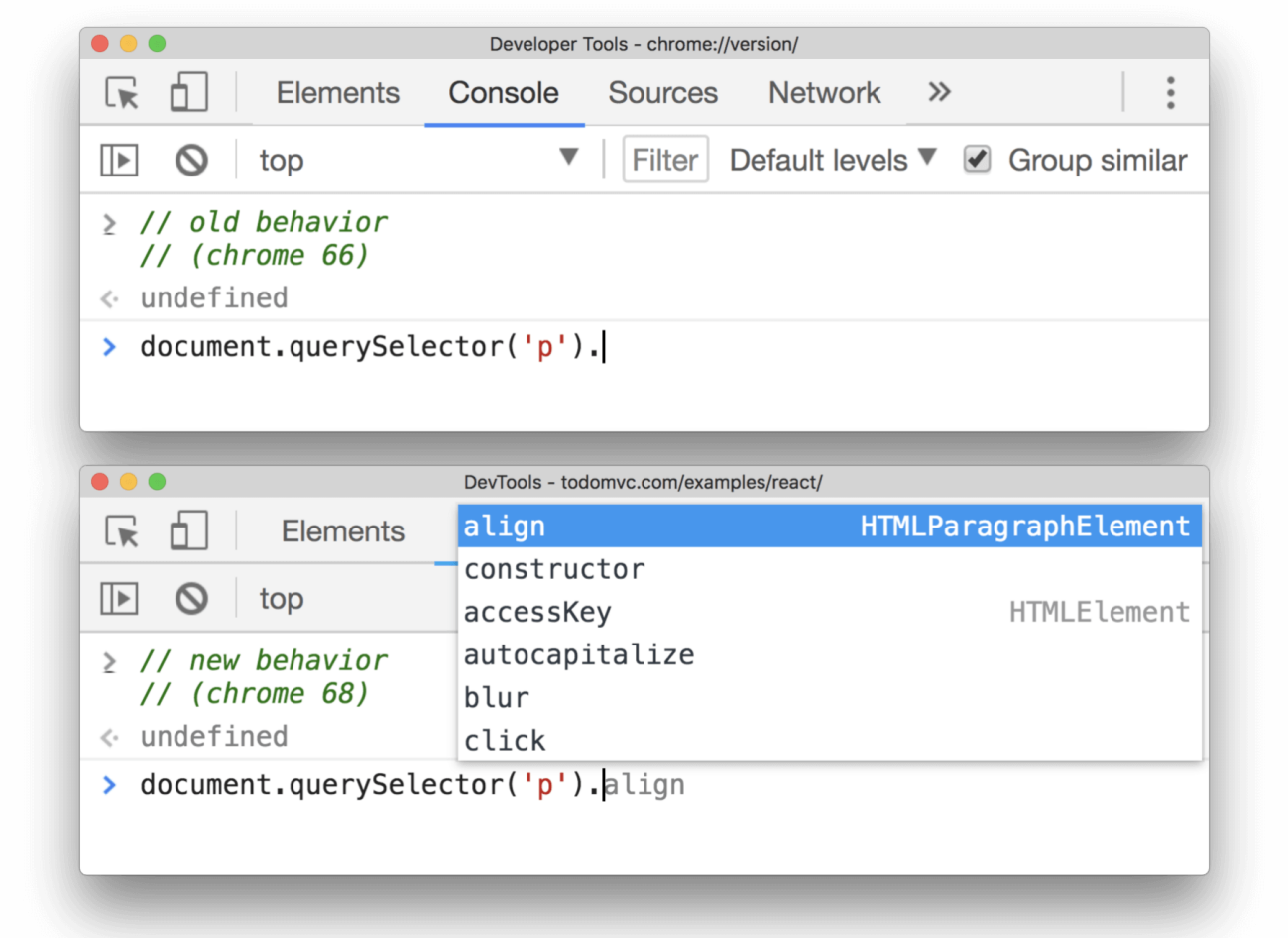

Figure 3. The top screenshot represents the old behavior, and the bottom screenshot represents the new behavior that supports function autocompletion

ES2017 keywords in autocomplete

ES2017 keywords, such as `await`, are now available in the Console's autocomplete UI.

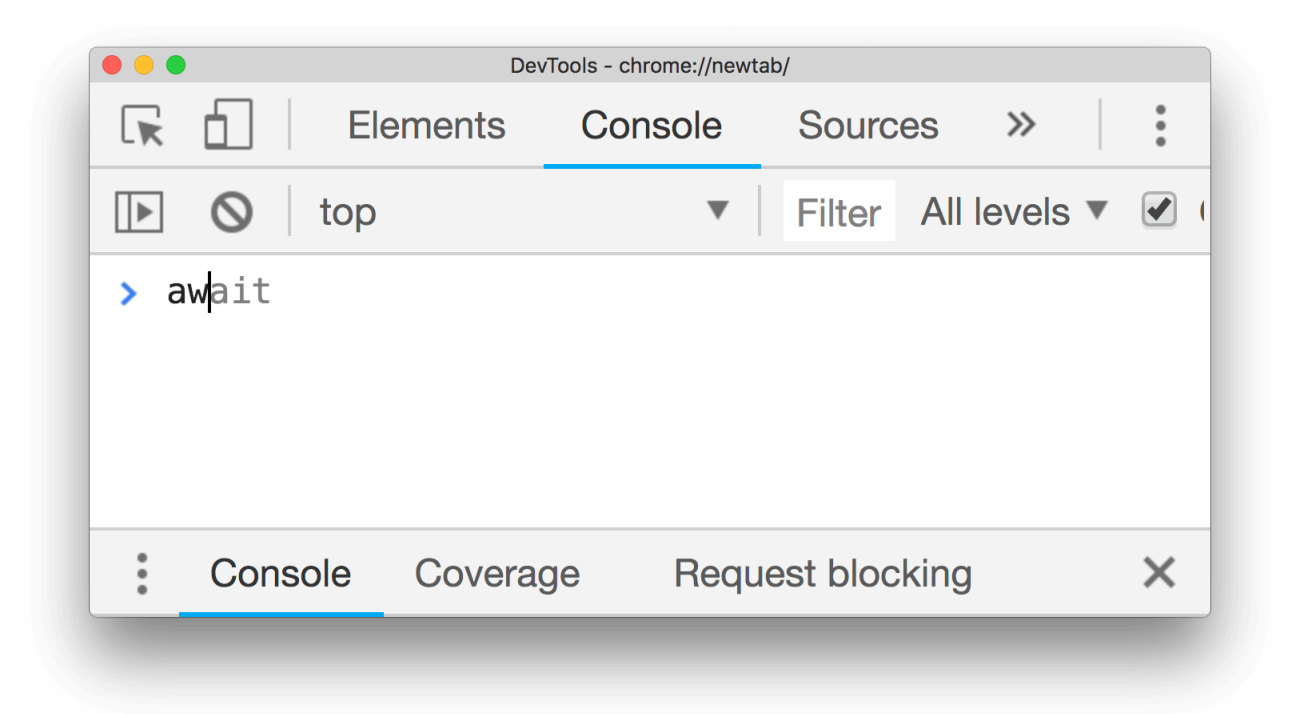

Figure 4. The Console now suggests `await` in its autocomplete UI

Faster, more reliable audits, a new UI, and new audits

Chrome 68 ships with Lighthouse 3.0. The next sections are a roundup of some of the biggest changes. See Announcing Lighthouse 3.0 for the full story.

Faster, more reliable audits

Lighthouse 3.0 has a new internal auditing engine, codenamed Lantern, which completes your audits faster, and with less variance between runs.

New UI

Lighthouse 3.0 also brings a new UI, thanks to a collaboration between the Lighthouse and Chrome UX (Research & Design) teams.

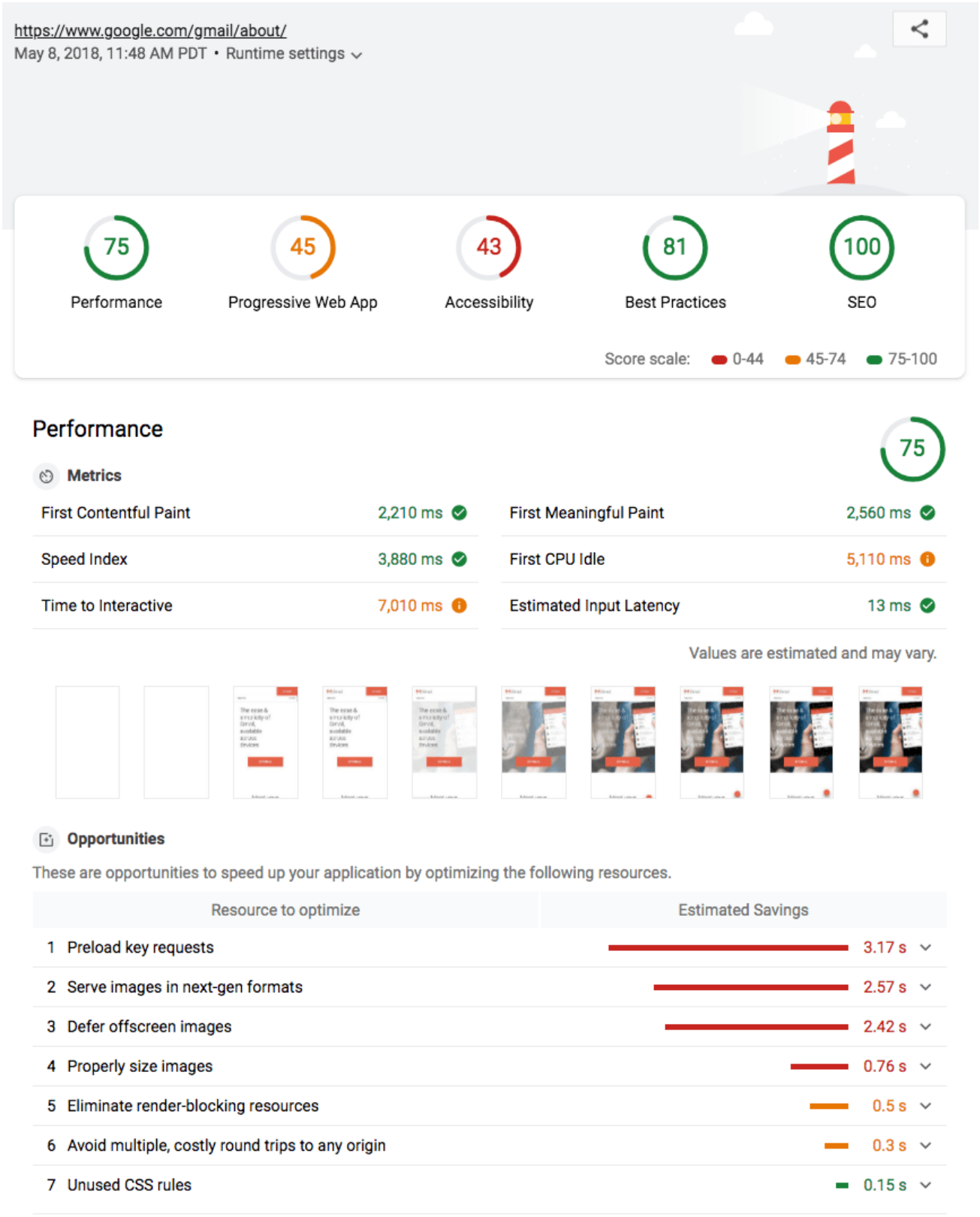

Figure 5. The new report UI in Lighthouse 3.0

New audits

Lighthouse 3.0 also ships with 4 new audits:

- First Contentful Paint
- robots.txt is not valid
- Use video formats for animated content
- Avoid multiple, costly round trips to any origin

BigInt support

Note: This isn't a DevTools features per se, but it is a new JavaScript capability that you can try out in the Console.

Chrome 68 supports a new numeric primitive called **`BigInt`**. **`BigInt`** lets you represent integers with arbitrary precision. Try it out in the Console:

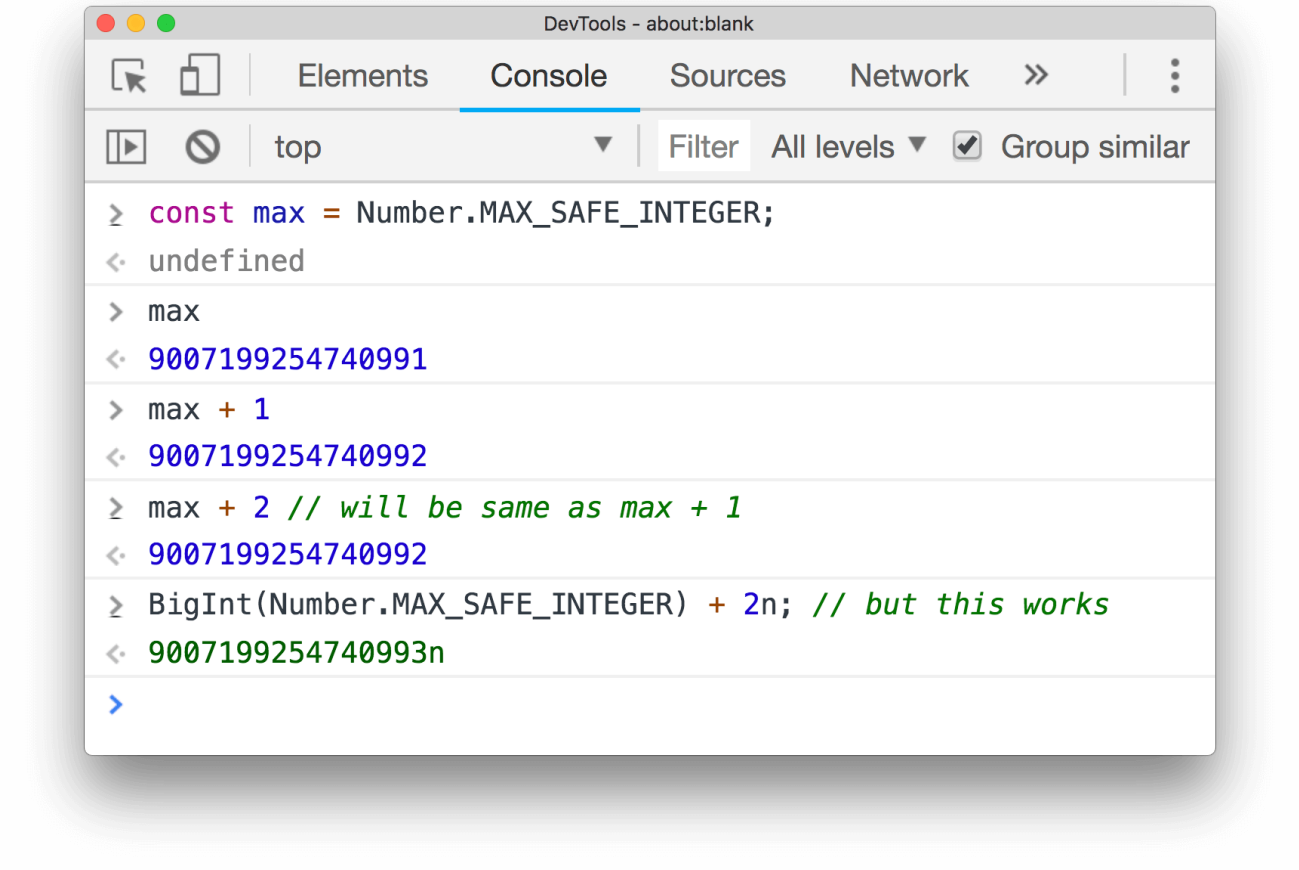

Figure 6. An example of **`BigInt`** in the Console

Add property path to watch

While paused on a breakpoint, right-click a property in the Scope pane and select **Add property path to watch** to add that property to the Watch pane.

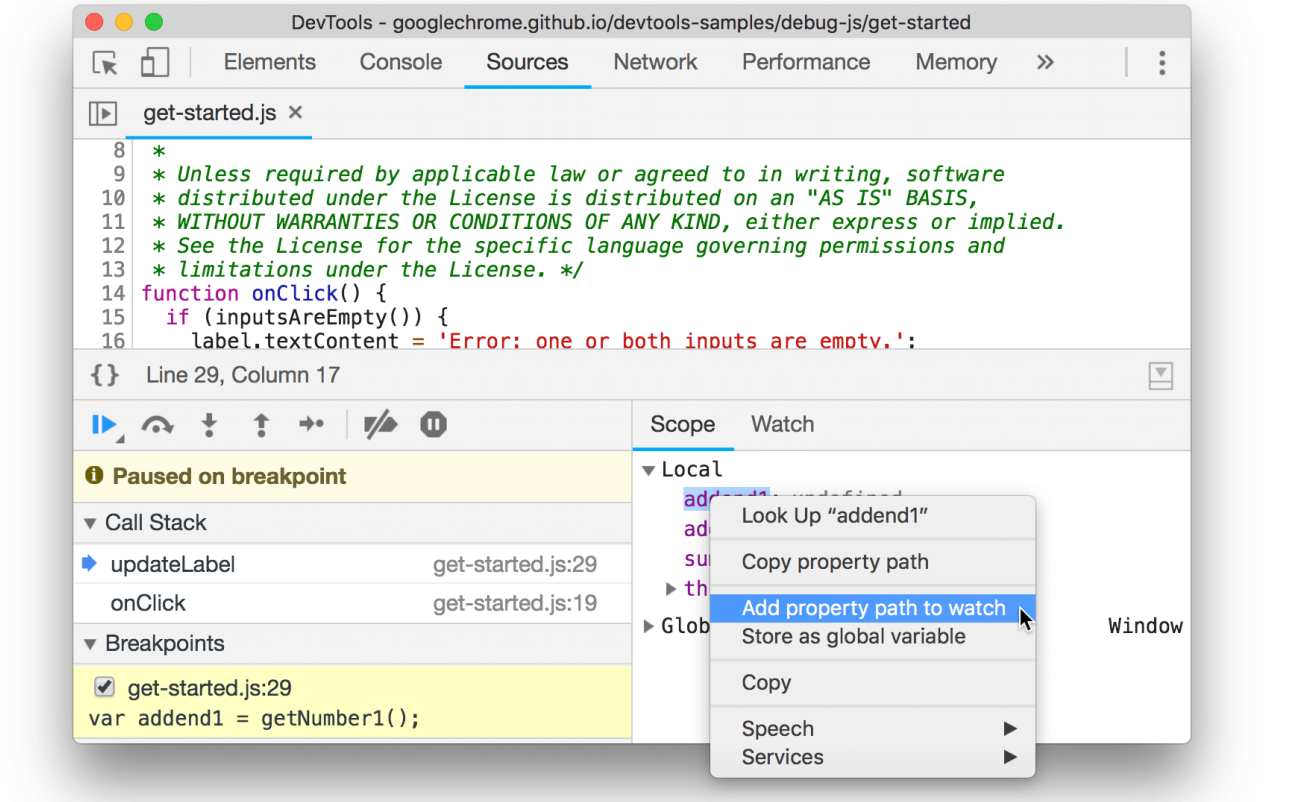

Figure 7. An example of **Add property path to watch**

"Show timestamps" moved to settings

The **Show timestamps** checkbox previously in **Console Settings** has moved to Settings.

Feedback

Was this page helpful?

YES

Great! Thank you for the feedback. Please use the feedback channels below to tell us what we're doing well, or how we can improve.

NO

Sorry to hear that. Please use the feedback channels below to tell us how we can improve.

To discuss the new features and changes in this post, or anything else related to DevTools:

- File bug reports at Chromium Bugs.
- Discuss features and changes on the Mailing List. Please don't use the mailing list for support questions. Use Stack Overflow, instead.
- Get help on how to use DevTools on Stack Overflow. Please don't file bugs on Stack Overflow. Use Chromium Bugs, instead.
- Tweet us at @ChromeDevTools.
- File bugs on this doc in the Web Fundamentals repository.

Consider Canary

If you're on Mac or Windows, please consider using Chrome Canary as your default development browser. If you report a bug or a change that you don't like while it's still in Canary, the DevTools team can address your feedback significantly faster.

Note: Canary is the bleeding-edge version of Chrome. It's released as soon as its built, without testing. This means that Canary breaks from time-to-time, about once-a-month, and it's usually fixed within a day. You can go back to using Chrome Stable while Canary is broken.

Previous release notes

See the devtools-whatsnew tag for links to all previous DevTools release notes.

Subscribe to our RSS or Atom feed and get the latest **updates** in your favorite feed reader!

Last updated July 17, 2018.