

Chrome DevTools

DevTools

Panels

More panels

Settings

Accessibility

Filter

Console

Overview

Understand errors and warnings better

Log messages

Run JavaScript

Watch JavaScript in real-time

Format and style messages

Feature reference

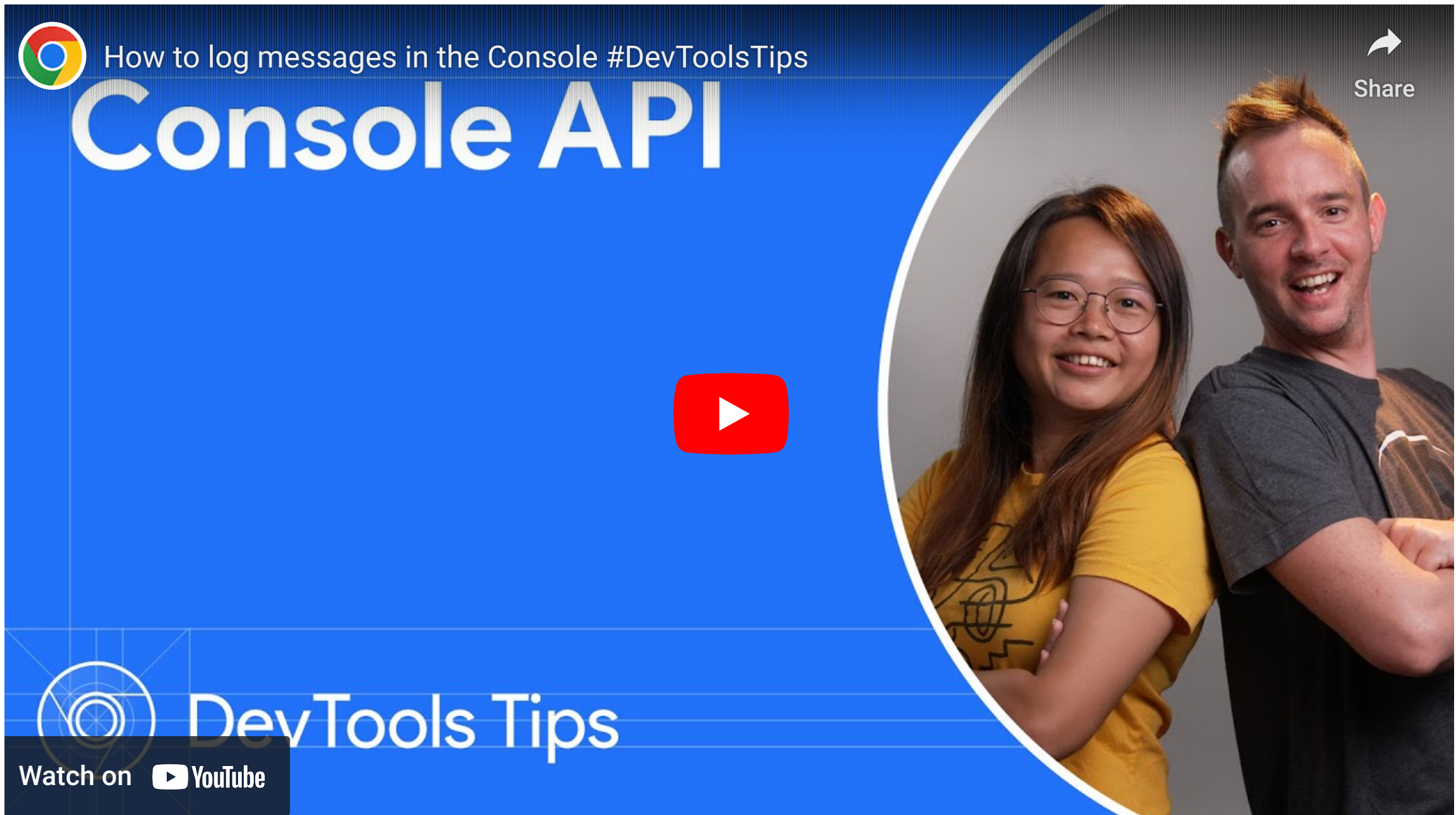
Home > Docs > Chrome DevTools > More panels

Was this helpful?

Console overview

Keyce Basques

This page explains how the Chrome DevTools Console makes it easier to develop web pages. The Console has 2 main uses: [viewing logged messages](#) and [running JavaScript](#).



Viewing logged messages

Web developers often log messages to the Console to make sure that their JavaScript is working as expected. To log a message, you insert an expression like `console.log('Hello, Console!')` into your JavaScript. When the browser executes your JavaScript and sees an expression like that, it knows that it's supposed to log the message to the Console. For example, suppose that you're in the process of writing the HTML and JavaScript for a page:

```
<!doctype html>
<html>
  <head>
    <title>Console Demo</title>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <script>
      console.log('Loading!');
      const h1 = document.querySelector('h1');
      console.log(h1.textContent);
      console.assert(document.querySelector('h2'), 'h2 not found!');
      const artists = [
        {
          first: 'René',
          last: 'Magritte'
        },
        {
          first: 'Chaim',
          last: 'Soutine'
        },
        {
          first: 'Henri',
          last: 'Matisse'
        }
      ];
      console.table(artists);
      setTimeout(() => {
        h1.textContent = 'Hello, Console!';
        console.log(h1.textContent);
      }, 3000);
    </script>
  </body>
</html>
```

Figure 1 shows what the Console looks like after loading the page and waiting 3 seconds. Try to figure out which lines of code caused the browser to log the messages.

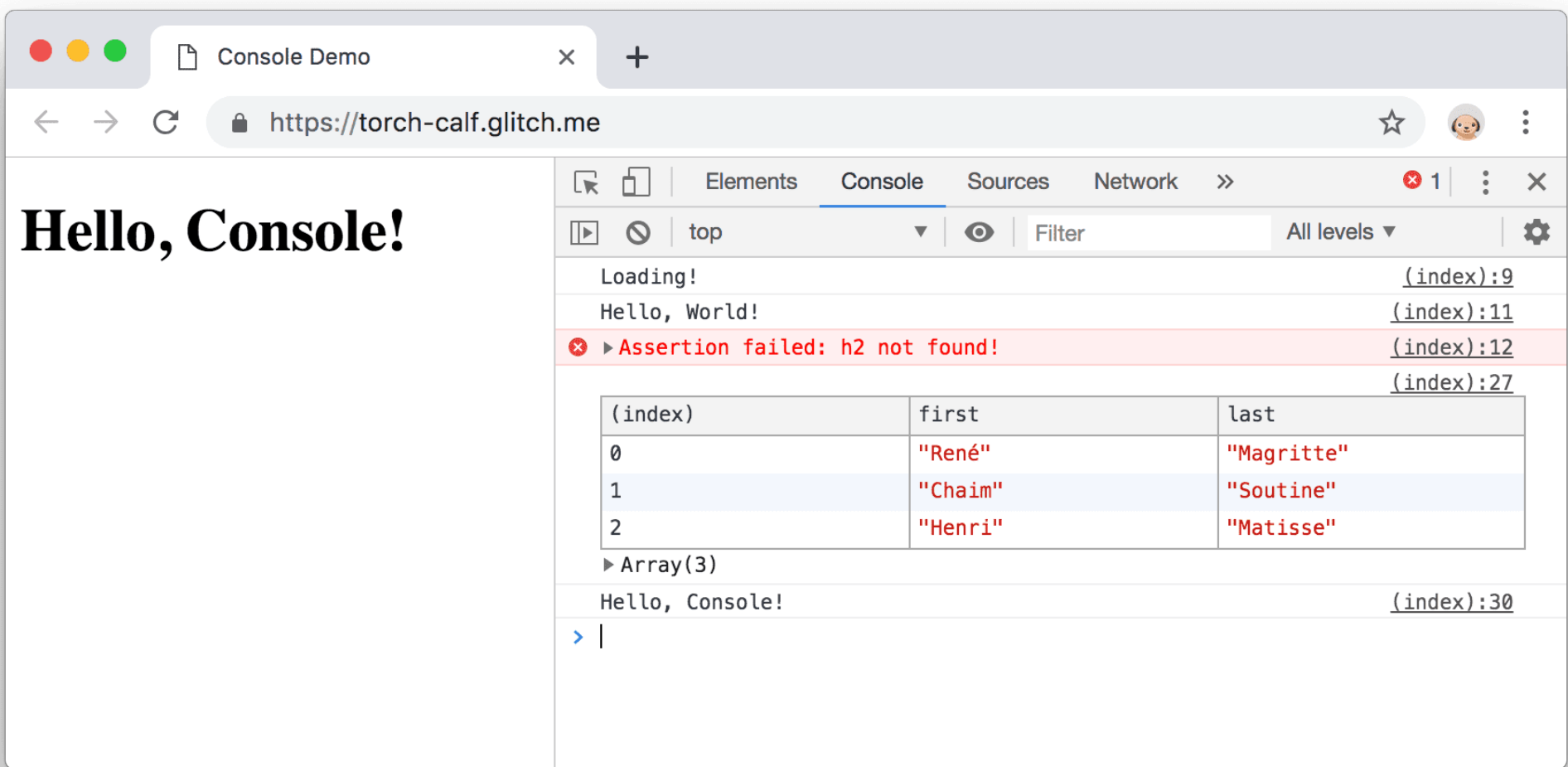


Figure 1. The Console panel.

Web developers log messages for 2 general reasons:

- Making sure that code is executing in the right order.
- Inspecting the values of variables at a certain moment in time.

See [Get Started With Logging Messages](#) to get hands-on experience with logging. See the [Console API Reference](#) to browse the full list of `console` methods. The main difference between the methods is how they display the data that you're logging.

Running JavaScript

The Console is also a [REPL](#). You can run JavaScript in the Console to interact with the page that you're inspecting. For example, **Figure 2** shows the Console next to the DevTools homepage, and **Figure 3** shows that same page after using the Console to change the page's title.

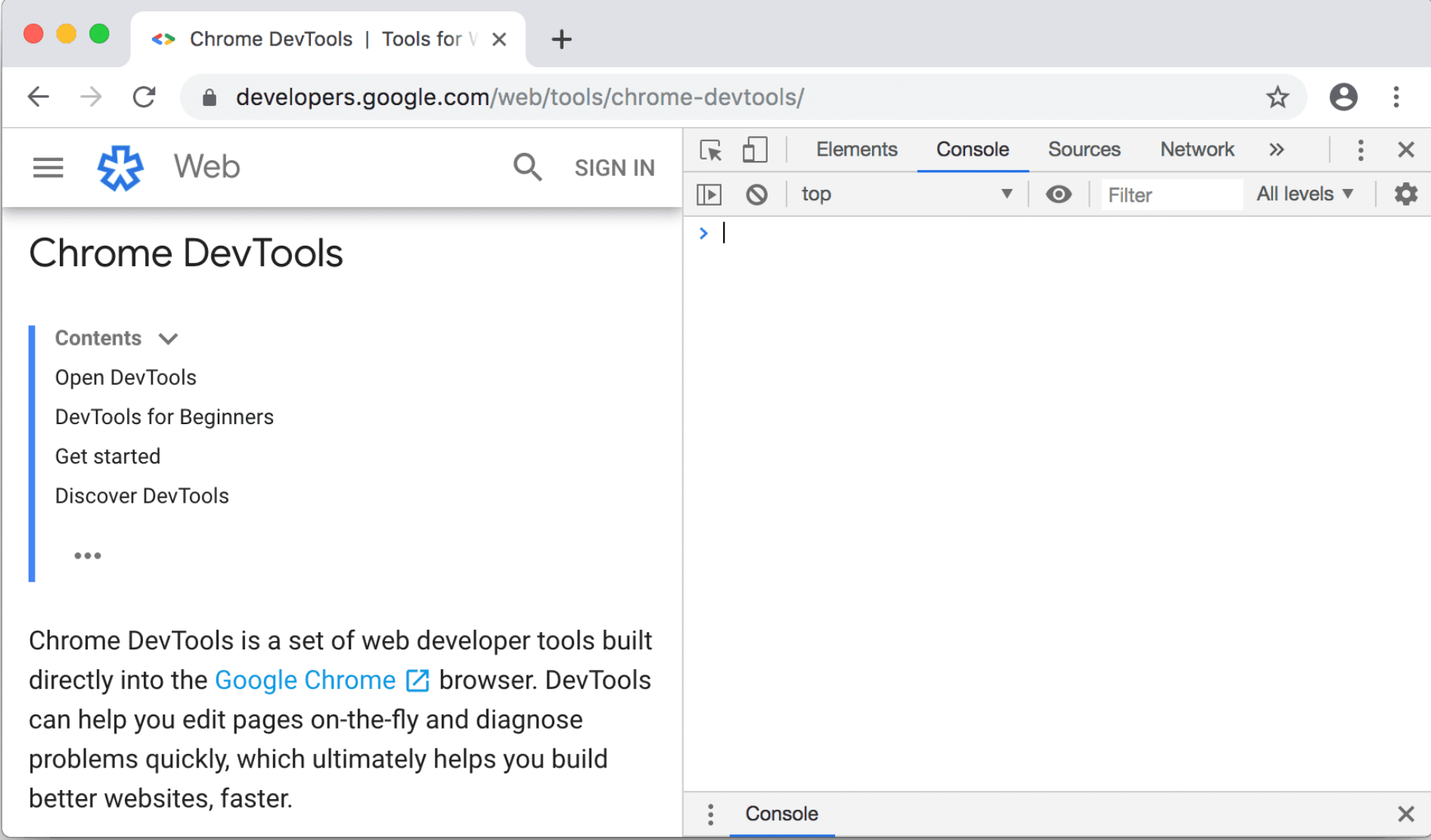


Figure 2. The Console panel next to the DevTools homepage.

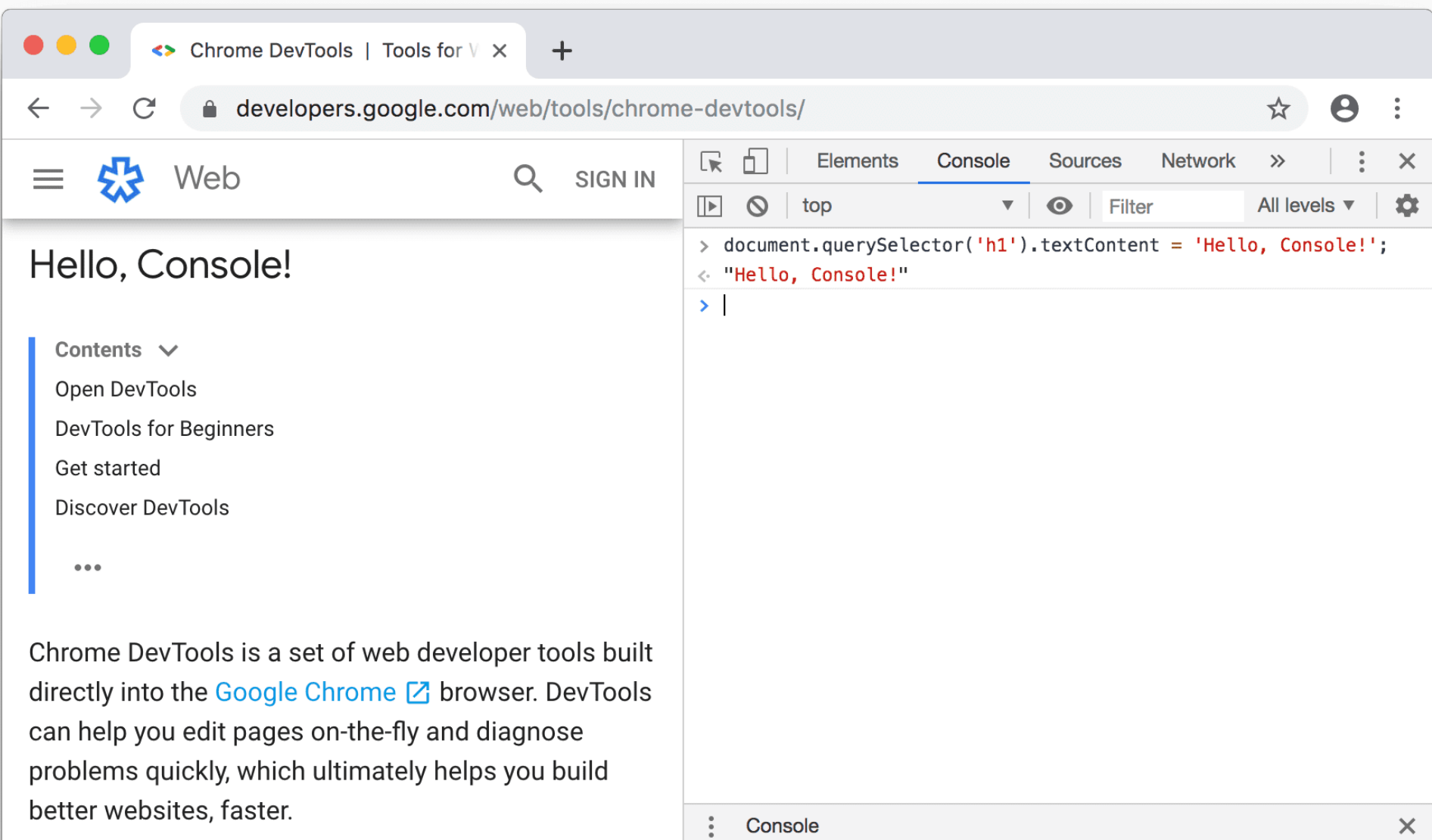


Figure 3. Using the Console to change the page's title.

Modifying the page from the Console is possible because the Console has full access to the page's `window`. DevTools has a few convenience functions that make it easier to inspect a page. For example, suppose that your JavaScript contains a function called `hideModal`. Running `debug(hideModal)` pauses your code on the first line of `hideModal` the next time that it's called. See [Console Utilities API Reference](#) to see the full list of utility functions.

When you run JavaScript you don't have to interact with the page. You can use the Console to try out new code that's not related to the page. For example, suppose you just learned about the built-in JavaScript Array method `map()`, and you want to experiment with it. The Console is a good place to try out the function.

See [Get Started With Running JavaScript](#) to get hands-on experience with running JavaScript in the Console.

Was this helpful?

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see the [Google Developers Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated 2019-04-18 UTC.

Contribute

File a bug

See open issues

Related content

Chromium updates

Case studies

Archive

Podcasts & shows

Connect

@ChromiumDev on X

YouTube

Chrome for Developers on LinkedIn