State of CSS 2021

- Feature usage and awareness
- Browser compatibility
- What's missing in CSS?
- Learn more

The State of CSS 2021 survey ran for the third year in a row, and reached over 8,000 developers worldwide. Let's look at some of the results and how they map to plans browsers have for adding CSS features in 2022.

Feature usage and awareness

For some features, there's a clear trend in usage and awareness year-over-year. CSS grid is an example of this.

Awareness of CSS Grid in State of CSS over time. The usage has grown from 55% to 83% between 2019 and 2021.

This matches the trend for CSS Grid in the 2021 Web Almanac and in Chrome's usage metrics. If you haven't used CSS Grid, now is a great time to learn it.

As more developers use grid, awareness of subgrid is also growing. Subgrid is already available in Firefox and will land in Chrome as part of the work by the Microsoft Team on GridNG.

Other features with strong growth in usage and awareness are `aspect-ratio`, scroll snap, and custom properties.

Browser compatibility

Browser compatibility is a common pain point for web developers and, to learn more about what CSS features cause the most issues, the survey asked, "Are there any CSS features you have difficulties using because of differences between browsers?"

Survey results for features with differences between browsers. The top three responses are grid, flexbox gap, and subgrid.

Survey results like these help guide prioritization for browser vendors. Many of the features are part of the Compat 2021 effort based on the MDN Browser Compatibility Report 2020, and others like subgrid are now being proposed as focus areas for Interop 2022.

What's missing in CSS?

The survey also asked, "Which feature would you most like to be able to use in CSS today?"

Survey results for the question what's missing in CSS. The top three responses are container queries, parent selector, and browser support.

Container Queries was the overall "winner", matching the 2020 results. Container queries enable component-based responsive design by allowing elements to query their parent selector for style changes. This is a much more local, specified scope than is currently possible with media queries.

Chrome is currently working on an experimental implementation, and funding the work of Miriam Suzanne, as she develops the specification in the CSS Working Group. You can try it out by going to `about:flags`, searching for, and enabling, the **Container Queries** flag in Canary, or by using the container query polyfill. You can learn more about container queries, as well as see some demos that use them in the latest series of Designing in the Browser or trying them yourself here.

https://www.youtube.com/watch?v=gCNMyYr7F6w

Learn more

Please see the full report to learn more, see recommended resources, or dig into the data yourself.

Thanks to Sacha Greif for running the survey, and to the over 8,000 web developers who generously took the time to answer it.

□□□ #3
□ □□□ □□□ 2026-10-31 22:18:59 UTC
□ cocobook-sync □□□ 2026-11-04 17:38:08 UTC