

Coming soon to CSS

Rachel Andrew

Gaps for everyone

The gap properties

gap, row-gap, column-gap

```
.flex {  
  display: flex;  
  margin: 0 -10px;  
}
```

```
.flex > * {  
  flex: 1 1 auto;  
  margin: 0 10px;  
}
```

```
.flex {  
  display: flex;  
  gap: 20px;  
}  
  
.flex > * {  
  flex: 1 1 auto;  
}
```

What happened to grid-gap?

Box alignment

The specification for alignment and space distribution across all layout methods.

The grid- properties are
maintained as aliases.

gap property for Flexbox - WD

☆ gap for flexbox containers to create gaps/gutters between flex items

Usage % of all users ?

- Global 75.53%
- U.K. 66.48%

Current aligned Usage relative Date relative Filtered All ⚙

IE	Edge *	Firefox	Chrome	Safari	Opera	Safari on iOS *	Opera Mini *	Android Browser *	Opera Mobile *	Chrome for Android	Firefox for Android	UC Browser for Android	Samsung Internet	QQ Browser	Baidu Browser	KaiOS Browser
	12-83	2-62	4-83		10-72											
6-10	84-90	63-88	84-90	3.1-14	73-75	3.2-14.4		2.1-4.4.4	12-12.1				4-13.0			
11	91	89	91	14.1	76	14.6	all	91	62	91	89	12.12	14.0	10.4	7.12	2.5
		90-91	92-94	15-TP												

Notes Test on a real browser Known issues (0) Resources (6) Feedback

Can we detect Flexbox gap support with **Feature Queries**?

```
@supports (display: grid) {  
  .grid {  
    /* grid things here */  
  }  
}
```

```
@supports (gap: 20px) {  
  .flex {  
    /* matches due to support  
for gap in grid. */  
  }  
}
```

::marker

```
li:first-child {  
    background-color: rgb(74, 112, 122);  
    color: #fff;  
}  
  
li:first-child::marker {  
    color: black;  
}
```

- List item one
- List item two
- List item three

```
h1 {  
  display: list-item;  
}
```

```
h1::marker {  
  content: "🧠";  
}
```


Veggies!

Veggies es bonus vobis, proinde vos postulo essum magis kohlrabi welsh onion daikon amaranth tatsoi tomatillo melon azuki bean garlic. Gumbo beet greens corn soko endive gumbo gourd. Parsley shallot courgette tatsoi pea sprouts fava bean collard greens dandelion okra wakame tomato. Dandelion cucumber earthnut pea peanut soko zucchini. Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea.

Aspect Ratios

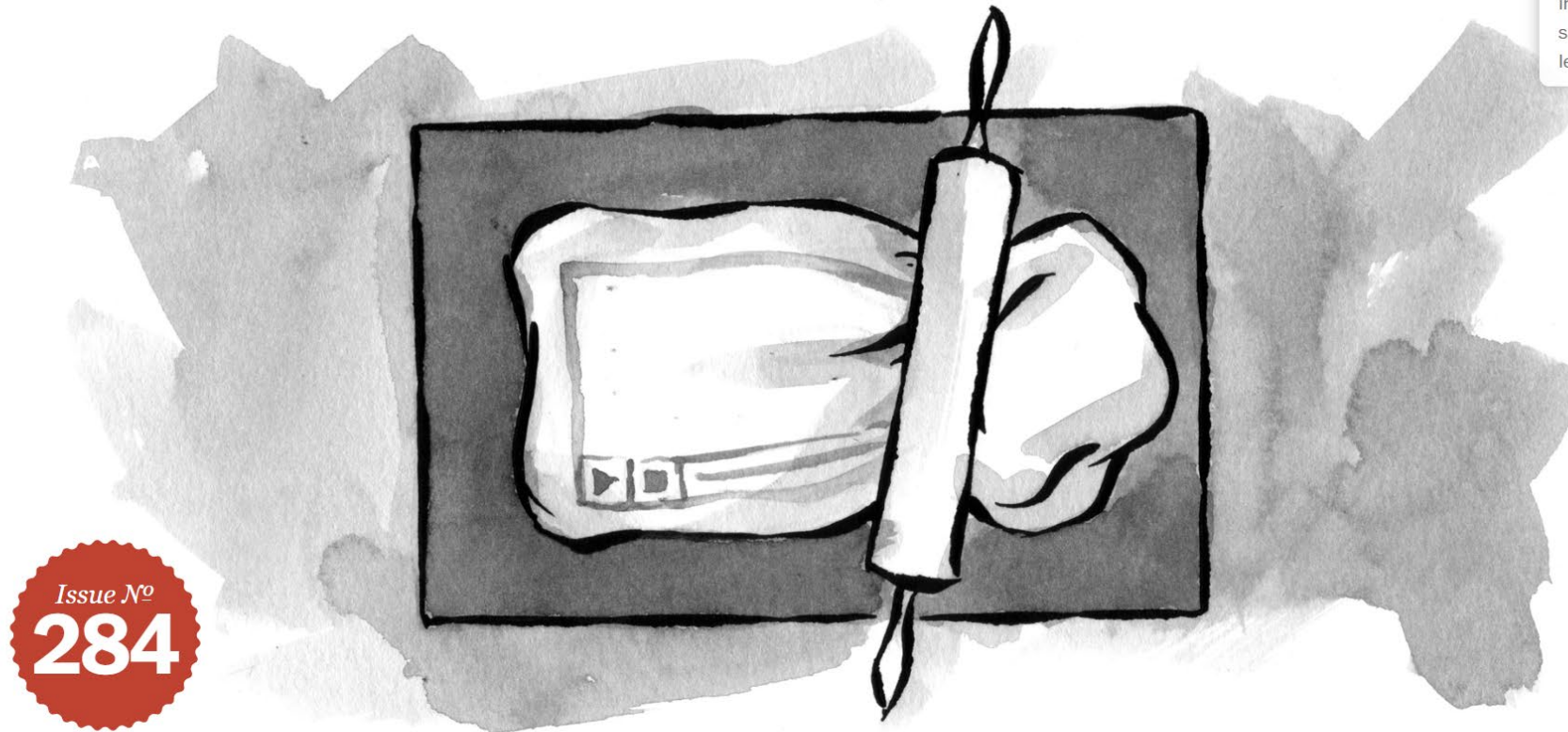


A LIST APART

Northwestern

INFORMATION DESIGN AND STRATEGY
School of Professional Studies

Northwestern's Online MS in Information Design and Strategy. Choose from tracks in content strategy, data science and analytics, and learning design.



Creating Intrinsic Ratios for Video

by [Thierry Koblentz](#) · May 25, 2009

Published in [CSS](#), [HTML](#), [JavaScript](#), [Responsive Design](#)

The aspect-ratio property

<https://developer.mozilla.org/en-US/docs/Web/CSS/aspect-ratio>

```
.grid {  
  display: grid;  
  gap: 20px;  
  grid-template-columns: repeat(auto-fill, minmax(250px, 1fr));  
}
```

```
.grid > * {  
  aspect-ratio: 1 / 1;  
  transform: rotate(0.1turn);  
}
```

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

Internal mapping

Modern browsers use the width & height attributes to internally map image aspect ratios to prevent loading shifts.

New color functions

color-mix() and color-contrast()


```
li {  
  background-color: color-mix(in srgb, #34c9eb 10%, white);  
}
```

10% #34c9eb

40% #34c9eb

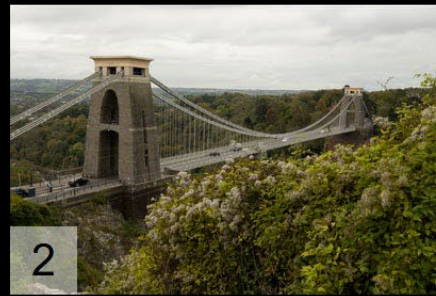
70% #34c9eb

```
li {  
  background-color: color-contrast(#008080 vs olive, var(--mycolor, #d2691e));  
}
```

Masonry in CSS

CSS Grid Level 3!


```
.container {  
  display: grid;  
  grid-template-columns: repeat(4, 1fr);  
  grid-template-rows: masonry;  
}
```

Use Firefox Nightly

Enable: `layout.css.grid-template-masonry-value.enabled`




ABOUT THE AUTHOR

Rachel Andrew is not only Editor in Chief of Smashing Magazine, but also a web developer, writer and speaker. She is the author of a number of books, including ... [More about Rachel Andrew ...](#)

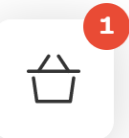
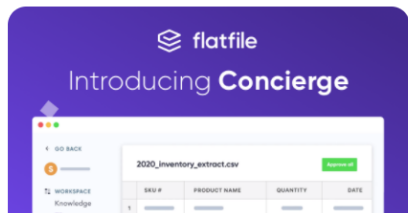
NOVEMBER 2, 2020 • [9 comments](#)

Native CSS Masonry Layout In CSS Grid

QUICK SUMMARY ↔ *There is now a specification for native CSS masonry layout, as part of the Grid Layout spec. In this article, Rachel Andrew explains how it works with the help of a couple of demos you can try out in Firefox Nightly.*

 9 min read [CSS](#), [CSS Grid](#), [Browsers](#) Share on [Twitter](#) or [LinkedIn](#)

A [LEVEL 3 OF THE CSS GRID SPECIFICATION](#) HAS BEEN PUBLISHED AS AN EDITOR'S Draft, this level describes a way to do Masonry layout in CSS. In this article, I'll explain the draft spec, with examples that you can try out in Firefox Nightly. While this is a feature you won't be able to use in production right now, your feedback would be valuable to help make sure it serves the requirements that you have for this kind of layout. So let's take a look.

<https://www.smashingmagazine.com/native-css-masonry-layout-css-grid/>

Subgrid

Subgrid examples

<https://gridbyexample.com/examples/#css-grid-level-2-examples>

... a team at Microsoft Edge are working on rearchitecting Chromium's Grid support to use the new LayoutNG engine - and as part of this are intending to add subgrid support!

<https://blog.chromium.org/2020/06/improving-chromiums-browser.html>

RenderingNG

<https://developer.chrome.com/blog/renderingng/>



Scott Jehl
@scottjehl



Wow... Okay, so here's something I've been trying to do for at least 10 years!

A CSS-only responsive table with fixed column & row headers, inside a layout, with scroll snapping! 🤖
codepen.io/scottjehl/pen/...

Works in: Firefox, Safari (iOS & Mac), Chrome (Mobile & Desktop), Edge

CSS responsive table with fixed column and row headers and scroll snap

		Col Header	Col Header	Col Header	Col Header	Col Header	Col Header	Col Header	Col Header
sidebar	Row Header	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data
	Row Header	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data
	Row Header	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data
	Row Header	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data
	Row Header	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data
	Row Header	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data
	Row Header	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data
	Row Header	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data
	Row Header	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data
	Row Header	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data
	Row Header	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data
	Row Header	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data	Cell Data

CSS responsive table with fixed column and row headers and scroll snap

...

[codepen.io](#)

4:15 PM · Jun 22, 2021 · Twitter Web App

227 Retweets 30 Quote Tweets 1,377 Likes



prefers_reduced_data

```
@media (prefers-reduced-data: no-preference) {  
  .hero {  
    /* download large image */  
  }  
}
```


Elements

Console

Sources

Network

Performance

Memory

Application

Security

Lighthouse

<!DOCTYPE html>

<html lang="en" class="enhanced js wf-loaded-stage1 wf-loaded-stage2">

<head>...</head>

<body class="homepage enhanced js anonymous non-subscriber" id="home">

<div class="cookies hidden" id="cookiebanner" lang="en" dir="ltr" name="cookiebanner" style="display: block;">...</div>

<div data-handler="PreLoad" id="top">...</div>

Skip to main content

Jump to list of all articles

<header class="main-header">...</header>

<div>...</div>

<div class="constrast-switch-container">...</div>

<main id="main" tabindex="-1">

<div class="featured-articles-container">

<h1 class="featured-articles__teaser" id="featured-articles__heading">...</h1>

<article class="featured-article">...</article>

... loaded-stage1.wf-loaded-stage2 body#home.homepage.enhanced.js.anonymous.non-subscriber main#main div.featured-articles-container article.featured-article h2.featured-article__title a.featured-article__title_a ...

Styles

Computed

Event Listeners

>>

Filter

:hov .cls +

element.style {

}

.featured-article__title main.css:7

.featured-article__title_a {

color: #fff;

text-shadow: none;

background: 0 0;

text-shadow:

1px 1px 1px rgba(0,0,0,.25);

position: static;

}

.featured-articles-container main.css:1

.author--white .author__image,

.featured-articles-container .featured-

Console

Rendering

×

Emulate a focused page

Emulates a focused page.

Emulate CSS media type

Forces media type for testing print and screen styles

No emulation

Emulate CSS media feature prefers-color-scheme

Forces CSS prefers-color-scheme media feature

No emulation

Emulate CSS media feature prefers-reduced-motion

Forces CSS prefers-reduced-motion media feature

No emulation

Emulate CSS media feature prefers-reduced-data

Forces CSS prefers-reduced-data media feature

No emulation

No emulation

prefers-reduced-data: reduce

No emulation

Slides & resources <https://noti.st/rachelandrew>

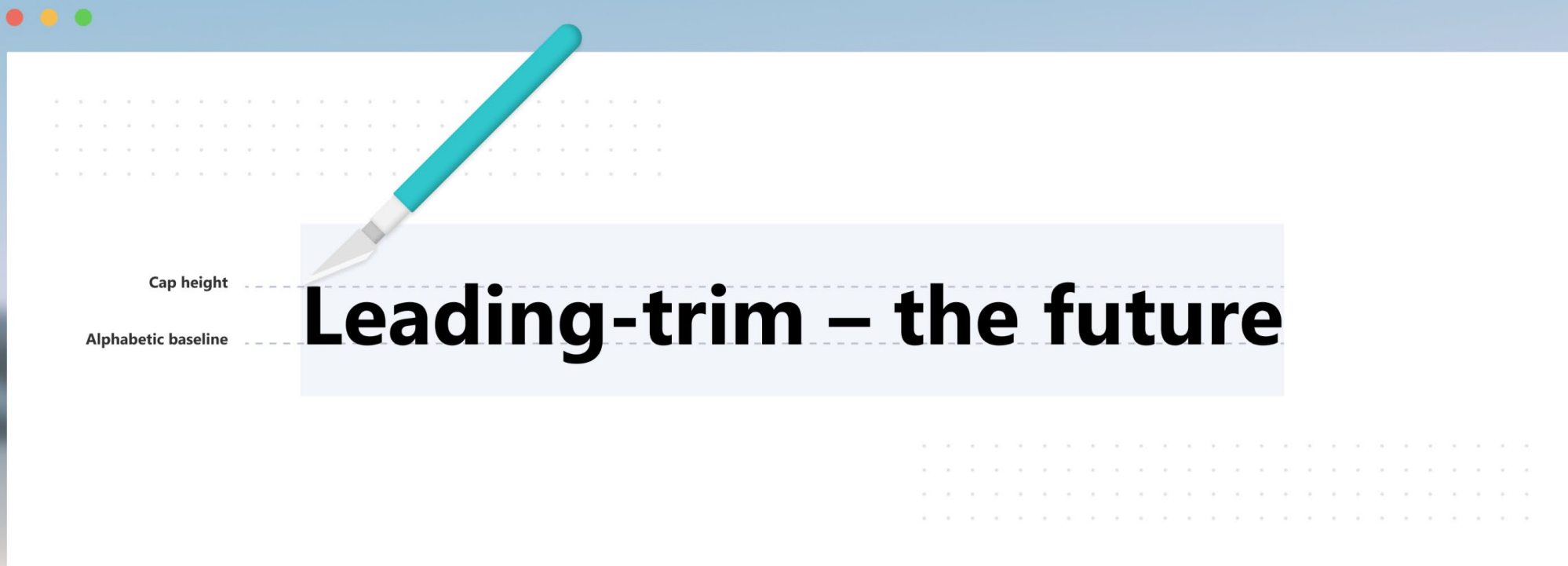
User preference media features

Part of Media Queries Level 5

leading-trim

Losing the **extra spacing** around text.

```
h1 {  
  text-edge: cap alphabetic;  
  leading-trim: both;  
}
```



Cap height

Alphabetic baseline

Leading-trim – the future

The problem with text boxes today

In a standard text box, there's almost always extra space above and below the actual text. Because of this, when you use a text box to measure and implement spacing, it ends up larger than you intended. The bigger the line height, the bigger the problem. The example below shows distances between text boxes set to 32px. As you can see, all the vertical spacing is visually much bigger than 32px, even though you set them all to the same value.

<https://medium.com/microsoft-design/leading-trim-the-future-of-digital-typesetting-d082d84b202>

Solving font layout shifts

size-adjust, ascent-override, descent-override, line-gap-override

```
@font-face {
  font-family: "fallback";
  src: local(Arial);
  size-adjust: 90%;
  ascent-override: 110%;
  descent-override: 110%;
  line-gap-override: 99%;
}

@font-face {
  font-family: "Webfont";
  src: url(/path/to/webfont.woff2) format('woff2');
}

html {
  font-family: "Webfont", "fallback";
}
```

Container Queries?

On their way!


```
.wide-column,  
.narrow-column{  
  contain: layout inline-size;  
}  
  
@container (min-width: 700px) {  
  .card {  
    grid-template-columns: 1fr 2fr;  
    grid-template-rows: auto 1fr;  
  }  
}
```


Anyone can contribute to specifications

You don't need to be an Invited Expert or representative.

Navigation

Web Platform Governance

Intellectual Property Rights
Management

Skills

Step by step guide to fix a W3C
or WHATWG bug

Paths For Growth

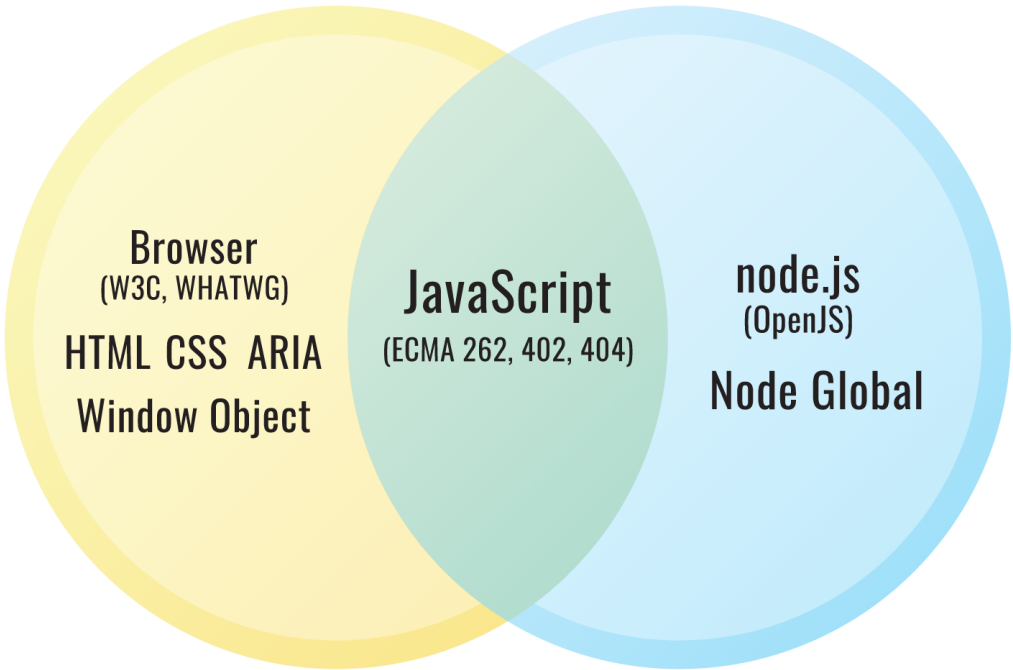
Case Study Examples

Interviews

Finding who to talk to

Glossary of Terms and Jargon

Getting started with a history, skill guide and how-to of web standards.



Introduction

The “web platform” is comprised of a collection of standardized **application programming interfaces (APIs)** that programmers use to make web pages and web applications. This includes technologies, like HTML, CSS and JavaScript. In addition to including a full implementation of the JavaScript programming language that web browsers share with **Node.js**, web browsers expose their own standardized **APIs** specifically made for interacting with web pages.

The web platform is powerful and expressive, but it’s also a little broken in parts. That’s where the other part of the web platform comes in. The term “web platform” *also* describes the people and practices that are meant to include everyone in improving the standard programming interfaces. The web can only work if it’s built with an understanding of all its constituents—the whole world! This

<https://wpc.guide/>

Read and **comment** on specification issues

<https://github.com/w3c/csswg-drafts/issues>

[css-grid] Ability to target items in nth-row of autofilled, implicit grid #1943

Edit

New issue

📄 Open

rachelandrew opened this issue on 6 Nov 2017 · 14 comments



rachelandrew commented on 6 Nov 2017

Member + 😊 ...

I've had various people ask me the same question as raised here: [rachelandrew/cssgrid-ama#115](https://github.com/rachelandrew/cssgrid-ama/issues/115)

The use case is as described in that issue. If the author creates the following grid:

```
nav {
  display: grid;
  grid-template-columns: repeat(auto-fill, minmax(150px, 1fr));
}
```

They then want to style items on the second (or nth) row differently to the first, they don't have a way to target it. I've mocked it up using MQs. <https://codepen.io/rachelandrew/pen/yPJKeo?editors=1100>

It's really a selectors issue, would the table pseudo-class selectors (<https://drafts.csswg.org/selectors-4/#table-pseudos>) work on grid items, or is this another instance of needing to be able to target a row or column of the grid, with the added complication of wanting to target the children of that row or column?

At the moment to do so, they need to go back to taking more control over the grid, rather than using auto-fill, so that they know where each item sits and can use nth-child on the items, or use media queries which makes for less flexible components.

👍 34

Assignees



No one—assign yourself

Labels



css-grid-3

Projects



None yet

Milestone



No milestone

Notifications

Customize

🔊 Unsubscribe



You're receiving notifications because you're watching this repository.

15 participants



Show use cases

As with any software development, real use cases are valuable.

  rachelandrew added the **Agenda+ F2F** label on 27 Feb



css-meeting-bot commented on 28 Feb • edited by frivoal ▼

Member +  ...

The CSS Working Group just discussed `column-span`, and agreed to the following:

- RESOLVED: Clarify the spec using the notion that this is an out of flow that leaves a forced break behind
- The full IRC log of that discussion

It will probably take longer
than you think for changes
to be made!

Patience is required for web platform contributions.

Raise browser bugs

Fix bugs, or request that features are implemented.

Learn to create a Reduced Test Case

★ Starred by 326 users

Owner: ----

CC: jfernando@igalia.com
r...@igalia.com
svil...@igalia.com

Status: Available (Open)

Components: [Blink>Layout>Grid](#)

Modified: 11 days ago

Editors: ----

EstimatedDays: ----

NextAction: ----

Notes: ----

OKR_link: ----

OS: [All](#)

Pri: [3](#)

Type: [Feature](#)

Hotlist-Recharge-Cold

Blocking: [Issue 79180](#)



Your Hotlists:  Update your hotlists

Other Hotlists: [css-grid](#)
[hotlist](#)
[Watch](#)

Issue 618969: [css-grid] Implement subgrid support

Reported by r...@igalia.com on Fri, Jun 10, 2016, 12:24 PM GMT+2


Project Member

 Code 

There's a new reduced subgrid proposal that we could try to implement at some point.

The spec section: <https://drafts.csswg.org/css-grid/#subgrids>

Just reporting the bug to keep track of it and use it as meta-bug during the implementation.

[Comment 1](#) by m...@mko.io on Sun, Jun 12, 2016, 10:56 AM GMT+2 

This makes grid really useful!

[Comment 2](#) by sheriffbot@chromium.org on Mon, Jun 12, 2017, 5:05 PM GMT+2

Project Member

Status: Untriaged (was: Available)

Labels: Hotlist-Recharge-Cold

This issue has been Available for over a year. If it's no longer important or seems unlikely to be fixed, please consider closing it out. If it is important, please re-triage the issue.

Sorry for the inconvenience if the bug really should have been left as Available. If you change it back, also remove the "Hotlist-Recharge-Cold" label.


For more details visit <https://www.chromium.org/issue-tracking/autotriage> - Your friendly Sheriffbot

[Comment 3](#) by e...@chromium.org on Mon, Jun 12, 2017, 7:55 PM GMT+2

Project Member

Status: Available (was: Untriaged)

Labels: -Type-Bug Type-Feature

[Comment 4](#) by svil...@gmail.com on Fri, Jun 23, 2017, 5:39 PM GMT+2 

It's just simply delayed. Subgrids were moved to Level 2 of the specs, so they are not a priority at all ATM. But it'll be eventually implemented.

[Comment 5](#) by sheriffbot@chromium.org on Mon, Jun 25, 2018, 6:04 PM GMT+2

Project Member

Status: Untriaged (was: Available)

This issue has been Available for over a year. If it's no longer important or seems unlikely to be fixed, please consider closing it out. If it is important, please re-triage the issue.

Sorry for the inconvenience if the bug really should have been left as Available.

Test and offer feedback on features behind flags.

This is the **best time** to make changes to a spec.

Incomplete List of Mistakes in the Design of CSS

That should be corrected if anyone invents a time machine. :P

- `white-space: nowrap` should be `white-space: no-wrap`
 - and line wrapping behavior should not have been added to `white-space`
- `vertical-align` should not apply to table cells. Instead the CSS3 alignment properties should exist in Level 1.
- `vertical-align: middle` should be `text-middle` or `x-middle` because it's not really in the middle, and such a name would better describes what it does.
- Percentage heights should be calculated against `fill-available` rather than being undefined in auto situations.
- Table layout should be sane.
- Box-sizing should be `border-box` by default.
- `background-size` with one value should duplicate its value, not default the second one to `auto`. Ditto `translate()`.
- `background-position` and `border-spacing` (all 2-axis properties) should take `*vertical*` first, to match with the 4-direction properties like `margin`.
- The 4-value shorthands like `margin` should go counter-clockwise (so that the inline-start value is before the block-start value).
- `z-index` should be called `z-order` or `depth` and should Just Work on all elements (like it does on flex items).
- `word-wrap/overflow-wrap` should not exist. Instead, `overflow-wrap` should be a keyword on `'white-space'`, like `nowrap` (`no-wrap`).
- The top and bottom margins of a single box should never have been allowed to collapse together automatically as this is the **root of all margin-collapsing evil**.
- Partial collapsing of margins instead of weird rules to handle min/max-heights?
- Tables (like other non-blocks, e.g. flex containers) should form pseudo-stacking contexts.
- The `currentcolor` keyword should have a dash, `current-color`. Likewise all other color multi-word keyword names.
- There should have been a predictable color naming system instead of arbitrary X11 names.
- `border-radius` should have been `corner-radius`.
- Absolutely-positioned replaced elements should stretch when opposite offset properties (e.g. `left+right`) are set, instead of being start-aligned.
- The `hyphens` property should be called `hyphenate`. (It's called `hyphens` because the XSL:FO people objected to `hyphenate`.)
- `rgba()` and `hsla()` should not exist, `rgb()` and `hsl()` should have gotten an optional fourth parameter instead (and the alpha value should have used the same format as R, G, and B or S and L).

Web Platform Tests

Help us **test** the web platform

web- platform- tests

Navigation

[Test Suite Design](#)

[Running Tests](#)

[Writing Tests](#)

[Reviewing Tests](#)

[Project Administration](#)

Quick search

web-platform-tests documentation

The web-platform-tests project is a W3C-coordinated attempt to build a cross-browser test suite for [the Web-platform stack](#). Writing tests in a way that allows them to be run in all browsers gives browser projects confidence that they are shipping software which is compatible with other implementations, and that later implementations will be compatible with their implementations. This in turn gives Web authors/developers confidence that they can actually rely on the Web platform to deliver on the promise of working across browsers and devices without needing extra layers of abstraction to paper over the gaps left by specification editors and implementors.

The most important sources of information and activity are:

- github.com/web-platform-tests/wpt: the canonical location of the project's source code revision history and the discussion forum for changes to the code
- web-platform-tests.org: the documentation website; details how to set up the project, how to write tests, how to give and receive peer review, how to serve as an administrator, and more
- web-platform-tests.live: a public deployment of the test suite, allowing anyone to run the tests by visiting from an Internet-enabled browser of their choice
- wpt.fyi: an archive of test results collected from an array of web browsers on a regular basis
- [Real-time chat room](#): the [IRC](#) chat room named `#testing` on irc.w3.org; includes participants located around the world, but busiest during the European working day; [all discussion is archived here](#)
- [Mailing list](#): a public and low-traffic discussion list

If you'd like clarification about anything, don't hesitate to ask in the chat room or on the mailing list.

Watch a Talk

If you prefer watching a video, here is a talk introducing web-platform-tests:

Comment on spec issues, raise new
issues, contribute examples, raise
browser bugs, write tests.

Thank you!