2018

@rachelandrew at Smashing Conf London

MALEZA / MALAS HIERBAS

Into the Weeds of CSS Layout

https://www.flickr.com/photos/protake/17383852206/

W3C Working Draft

TABLE OF CONTENTS

1 **Introduction**

2 **Subgrids**

2.1 Establishing a Subgrid: Dual-axis Proposal

2.2 Establishing a Subgrid: Per-Axis Proposal

2.3 Characteristics of a Subgrid Item

2.4 Subgrid Sizing Algorithm

3 **Aspect-ratio–controlled Gutters**

Conformance

Document conventions

Conformance classes

Requirements for Responsible Implementation of CSS

Partial Implementations

Implementations of Unstable and Proprietary Features

Implementations of CR-level Features

Index

Terms defined by this specification

Terms defined by reference

References

Normative References

Property Index

Issues Index

CSS Grid Layout Module Level 2

W3C First Public Working Draft, 6 February 2018

This version:

https://www.w3.org/TR/2018/WD-css-grid-2-20180206/

Latest published version:

https://www.w3.org/TR/css-grid-2/

Editor's Draft:

https://drafts.csswg.org/css-grid-2/

Issue Tracking:

Inline In Spec

GitHub Issues

Editors:

Tab Atkins Jr. (Google)

Elika J. Etemad / fantasai (Invited Expert)

Rossen Atanassov (Microsoft)

Abstract

This CSS module defines a two-dimensional grid-based layout system, optimized for user interface design. In the grid layout model, the children of a grid container can be positioned into arbitrary slots in a predefined flexible or fixed-size layout grid. Level 2 expands Grid by adding “subgrid” capabilities for nested grids to participate in the sizing of their parent grids; and aspect-ratio–controlled gutters.

CSS is a language for describing the rendering of structured documents (such as HTML and XML) on screen, on paper, in speech, etc.

Status of this document

How big is that box?

A LIST APART

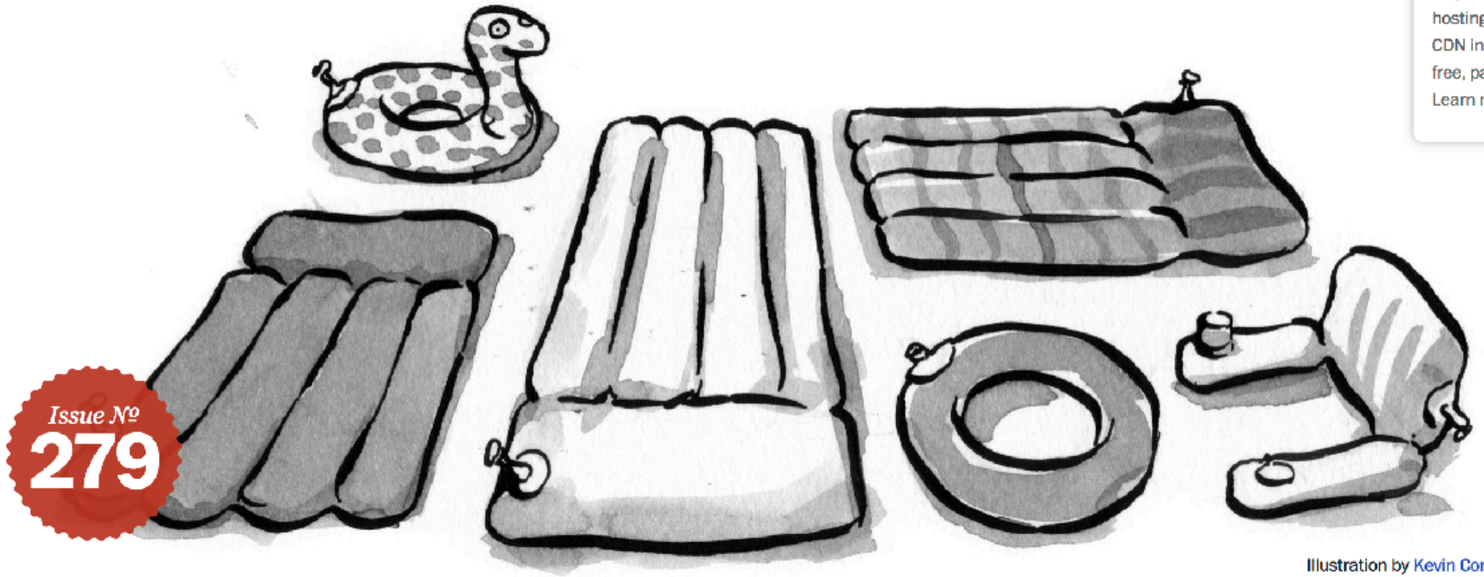


Illustration by Kevin Cornell

Fluid Grids

by **Ethan Marcotte** · March 03, 2009

Published in CSS, HTML, Graphic Design, Layout & Grids, Responsive Design

Early last year, I worked on the redesign of a rather content-heavy website. Design requirements were fairly light: the client asked us to keep the organization's existing logo and to improve the dense typography and increase legibility. So, early on in the design process, we spent a sizable amount of time planning a well-defined grid for a library of content modules.

https://twitter.com/devine_howest/status/959036561839960070

powered by: Devine howest

© 2018 Twitter About Help Center Terms
Privacy policy Cookies Ads info

Devine
@devine_howest

Devine - Digital Design & Development, a 3 year bachelor degree powered by howest.be

Kortrijk
devine.be
Joined January 2010

Devine
@devine_howest

Follow

Today we are celebrating International box-sizing Awareness Day. Anyone wants some padding? 🍰 #devinehowest @chriscoyier @Real_CSS_Tricks @paul_irish css-tricks.com/international- …

4:11 AM - 1 Feb 2018 from Kortrijk, België

25 Retweets **58** Likes

3 25 58

css layout frameworks

All Images Videos News Maps More Settings Tools

About 12,200,000 results (0.60 seconds)

Bulma: a modern CSS framework based on Flexbox

https://bulma.io/

A single element for a Metro UI-style **CSS grid**. Vertical... Top tile ...tiles. Bottom tile. Middle tile. With an image. Wide tile. Aligned with the right tile. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin ornare magna eros, eu pellentesque tortor vestibulum ut. Maecenas non massa sem. Etiam finibus odio quis feugiat ...

Docs · Breadcrumb · Box · Container

Skeleton: Responsive CSS Boilerplate

getskeleton.com/

You should use Skeleton if you're embarking on a smaller project or just don't feel like you need all the utility of larger **frameworks**. Skeleton only styles a handful of standard HTML elements and includes a **grid**, but that's often more than enough to get started. In fact, this site is built on Skeleton and has ~200 lines of custom ...

Pure CSS

https://purecss.io/

A set of small, responsive **CSS** modules that you can use in every web project.

Grids · Layouts · Get Started · Forms

What are Frameworks? 22 Best Responsive CSS Frameworks for Web ...

https://www.awwwards.com/what-are-frameworks-22-best-responsive-css-frameworks...

20 Feb 2013 - Front-end **Frameworks** (or **CSS Frameworks**). Frontend **frameworks** usually consist of a package made up of a structure of files and folders of standardized code (HTML, **CSS**, JS documents etc.) The usual components are: **CSS** source code to create a **grid**: this allows the developer to position the different ...

Best CSS Frameworks of 2017 | Three29

https://three29.com/best-css-frameworks-2017/

31 Mar 2017 - Allow us to lend a hand in sorting through the what's what in **CSS frameworks** in 2017. For those who are less tech savvy, a quick rundown. Building a website is a complicated process that involves a lot of moving parts. One of those components is the visual style and **layout** of a site. That's where our friend ...

https://www.smashingmagazine.com/2018/01/understanding-sizing-css-layout/

How Big Is That Box? Understar

https://www.smashingmagazine.com/2018/01/understanding-sizing-css-layout/

Search

A key feature of Flexbox and Grid Layout is that they can deal with distributing available space between, around and inside grid and flex items. Quite often this *just works*, and we get the result we were hoping for without trying very hard. This is because the specifications attempt to default to the most likely use cases. Sometimes, however, you might wonder why something ends up the size that it is. Or, you might want to do something different to the default behavior. To do so, you need to know something of how the underlying algorithms figure out how to distribute space.

In this article, I'm going to share with you some interesting things about sizing boxes in CSS. I've picked out a few things from the specifications that I believe are vital in terms of understanding exactly how big that box is. Take some time to read through, and I think you'll find sizing in Grid a lot less mysterious!

JANUARY 16, 2018 • 8 COMMENTS

How Big Is That Box? Understanding Sizing In CSS Layout

CSS 236 # Layouts 36 # Browsers 30

TABLE OF CONTENTS

01 Length Units
02 Percentages
03 CSS Intrinsic And Extrinsic Sizing
04 Content-Based Sizing In CSS Grid Layout
05 Auto-Sized Tracks
06 fr Units

TAKING A CLOSER LOOK AT BFC

Como de grande es un GRID ?

How big is a grid?

W3C Candidate Recommendation

CSS Grid Layout Module Level 1

396 Tests

W3C Candidate Recommendation, 14 December 2017

This version:

https://www.w3.org/TR/2017/CR-css-grid-1-20171214/

Latest published version:

https://www.w3.org/TR/css-grid-1/

Editor's Draft:

https://drafts.csswg.org/css-grid/

Previous Versions:

https://www.w3.org/TR/2017/CR-css-grid-1-20170509/
https://www.w3.org/TR/2016/WD-css-grid-1-20160519/
https://www.w3.org/TR/2015/WD-css-grid-1-20150917/
https://www.w3.org/TR/2015/WD-css-grid-1-20150806/
https://www.w3.org/TR/2015/WD-css-grid-1-20150317/
https://www.w3.org/TR/2014/WD-css-grid-1-20140513/
https://www.w3.org/TR/2014/WD-css-grid-1-20140123/
https://www.w3.org/TR/2013/WD-css3-grid-layout-20130402/
https://www.w3.org/TR/2012/WD-css3-grid-layout-20121106/

Test Suite:

http://test.csswg.org/suites/css-grid-1_dev/nightly-unstable/

Issue Tracking:

Disposition of Comments
Inline In Spec
GitHub Issues

Editors:

https://drafts.csswg.org/css-grid-1/#layout-algorithm

Cada pista tiene especificado minimo y un maximo funciones de dimensionado

“Each track has specified minimum and maximum sizing functions (which may be the same).”

el cual quizas sea el msimo

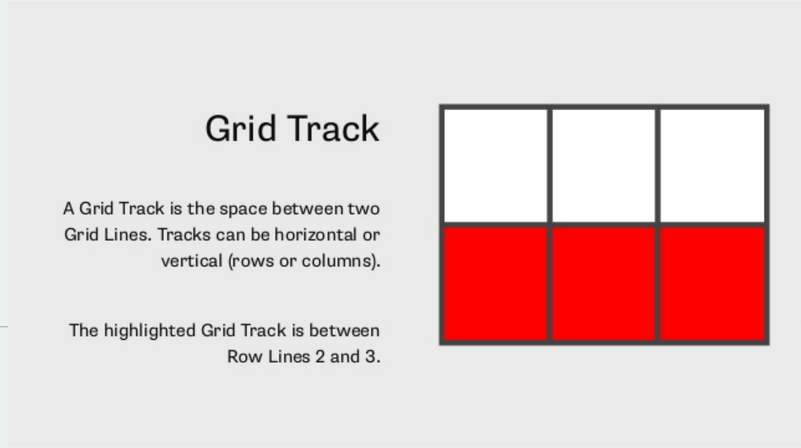
–11.1 Grid Sizing Algorithm

- Fixed sizing:
  lengths such as px or em, or a resolvable percentage
- An intrinsic sizing function
  auto, min-content, max-content, fit-content
- A flexible sizing function
  The fr unit

En CSS, el tamaño intrínseco de un elemento es el tamaño que se basaría en su contenido, si no se le aplicaran factores externos. Por ejemplo, los elementos en línea tienen un tamaño intrínseco: width, height y el margin vertical y el padding no tienen impacto, aunque el margin horizontal y el padding sí.

Intrinsic Sizing

auto



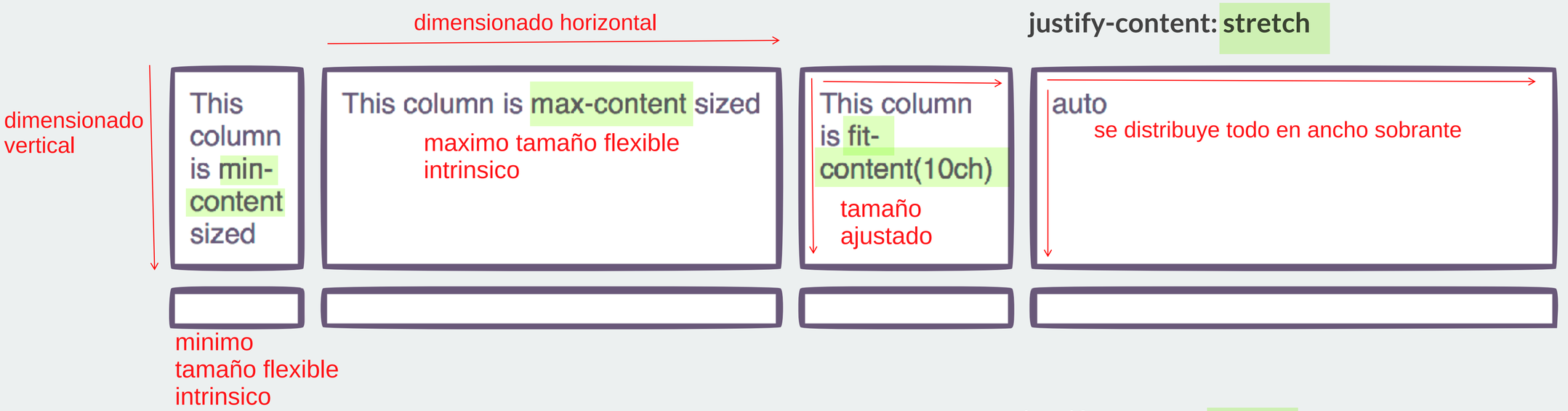
franja

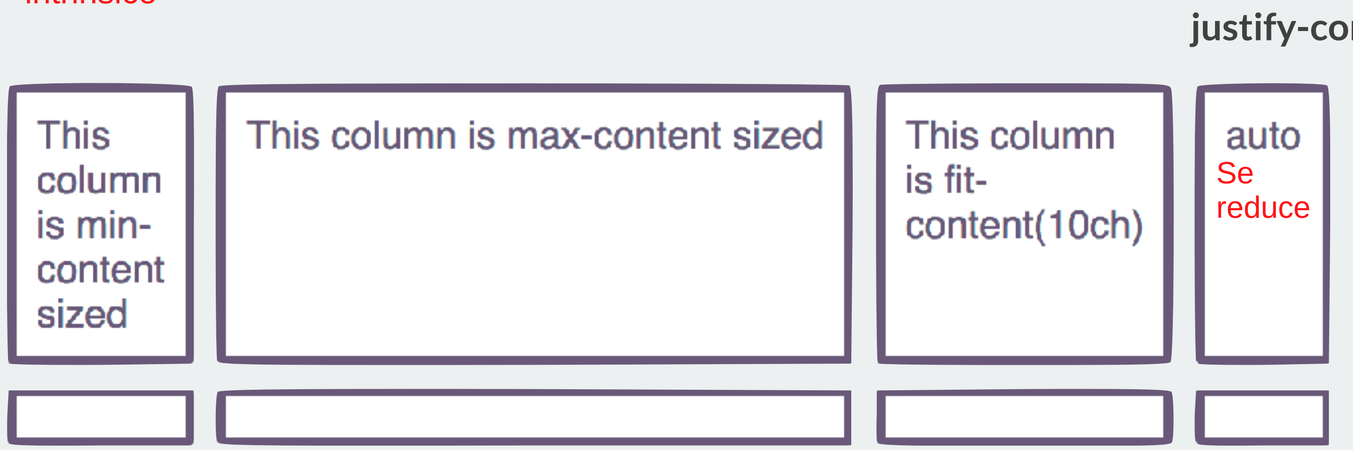
Default size of grid tracks. Tracks sized *auto* will stretch to take up space in the grid container.

para ocupar el espacio

Tamaño por defecto 'Grid Tracks'
'auto' estirara para ocupar todo el espacio del 'Grid-Container'

https://codepen.io/rachelandrew/pen/ppYPZR





https://codepen.io/rachelandrew/pen/ppYPZR

Intrinsic Sizing: auto

auto-dimensionado de la banda/franja estirara en la dirección de la linea

The auto-sized track will stretch in the inline direction.

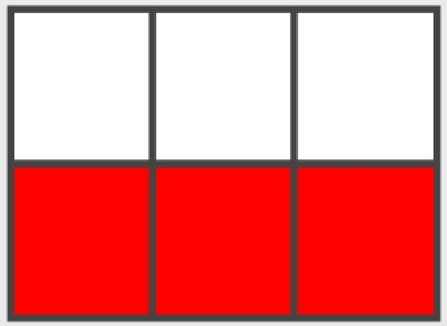
anular comportamiento estiramiento

Use justify-content: start to override the stretching behaviour.

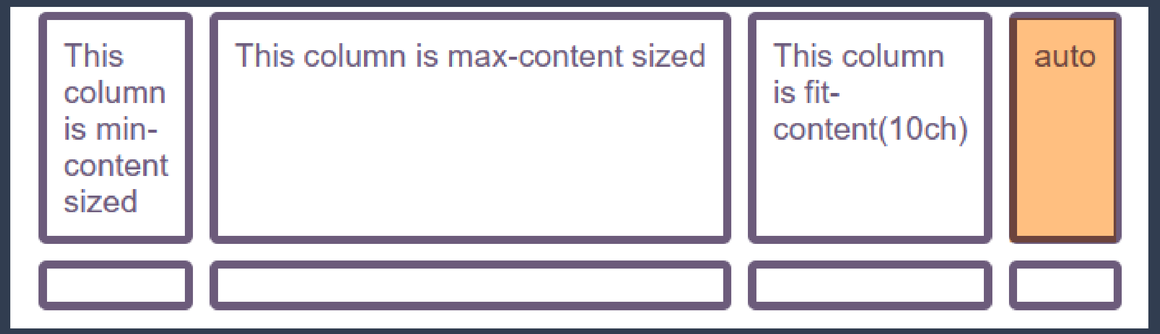
Grid Track

A Grid Track is the space between two Grid Lines. Tracks can be horizontal or vertical (rows or columns).

The highlighted Grid Track is between Row Lines 2 and 3.



```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns:
    min-content
    max-content
    fit-content(10ch)
    auto;
  justify-content: start;
}
```



Intrinsic Sizing

min-content

Tamaño más pequeño del elemento puede ser tomando ventaja

The smallest size the item can be, taking advantage of all soft-wrapping opportunities.

El tamaño más pequeño que el artículo puede ser, ventaja de todo el envoltorio blando oportunidades

Texto se ajusta hacia abajo del grid-item

Un hard-wrap - ajuste rígido inserta saltos de línea reales en el texto en los puntos de ajuste,
Un soft-wrap - con un ajuste suave el texto real todavía está en la misma línea pero parece que está dividido en varias líneas.

https://codepen.io/rachelandrew/pen/VyOpXj

reduce sin desbordamiento

Intrinsic Sizing: min-content

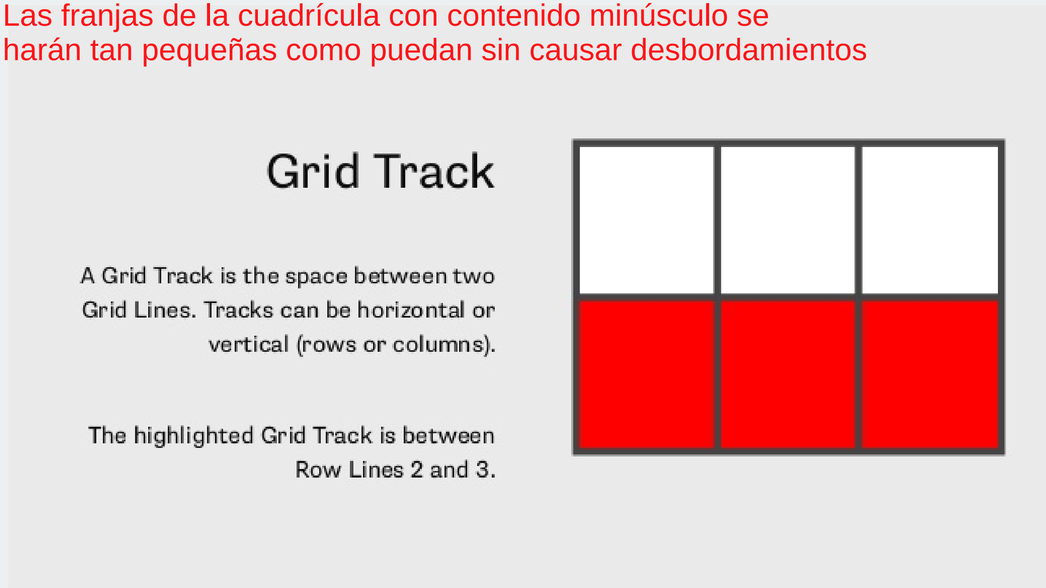
rejilla banda/franja dimensionado con "min-content" convertira tan pequeños como

Grid tracks sized with min-content will become as small as they can without causing overflows.

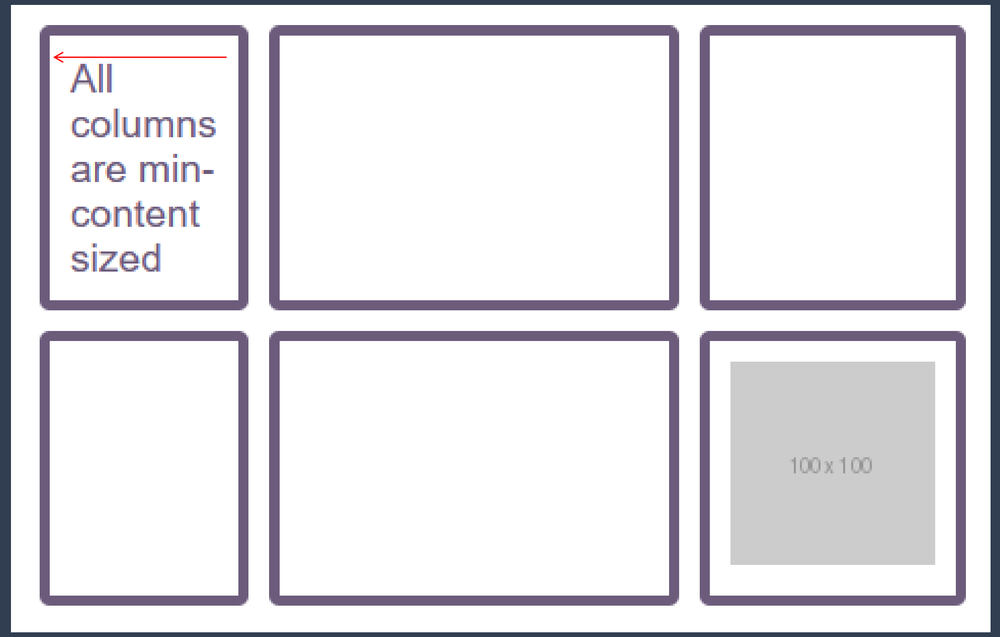
ellos pueden sin causa

desbordamiento

Las franjas de la cuadrícula con contenido minúsculo se harán tan pequeñas como puedan sin causar desbordamientos




```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns:
    min-content
    min-content
    min-content;
}
```



https://codepen.io/rachelandrew/pen/VyOpXj

EL CONTENIDO DEL GRID-TRACK DEFINE SU TAMAÑO SIEMPRE
BUSCANDO REDUCIRSE AL MAXIMO RESPECTO
A SU CONTENEDOR PRINCIPAL GRID-CONTAINER

The word ‘columns’ is defining the size of track 1.

A 100px image defines the size of track 3

This item has a width which defines the size of track 2

Este elemento tiene un ancho
el cual define el tamaño de la franja
banda 2

Intrinsic Sizing

max-content

El más grande tamaño de la franja puede ser

The largest size the track can be, no soft-wrapping will occur. Overflows may happen.

el texto no se se ajustara hacia abajo en varias lineas

desbordamiento quizas pasa

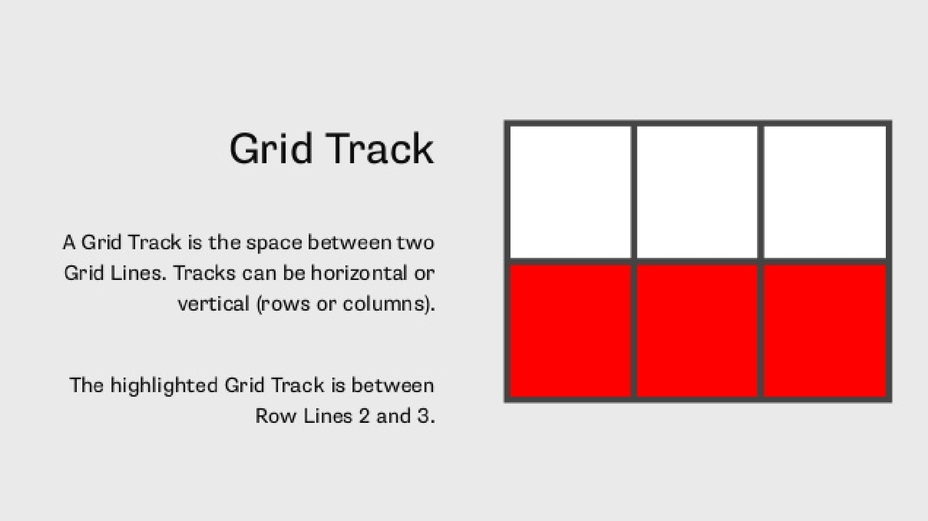
https://codepen.io/rachelandrew/pen/xpNdxV

Intrinsic Sizing: max-content

Grid tracks sized with max-content will become as large as needed, and may overflow the grid container.

se convertirá tan grande como

necesitado quizas desbordamiento del contenedor




```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns:
    max-content
    max-content
    max-content;
}
```

https://codepen.io/rachelandrew/pen/xpNdxV

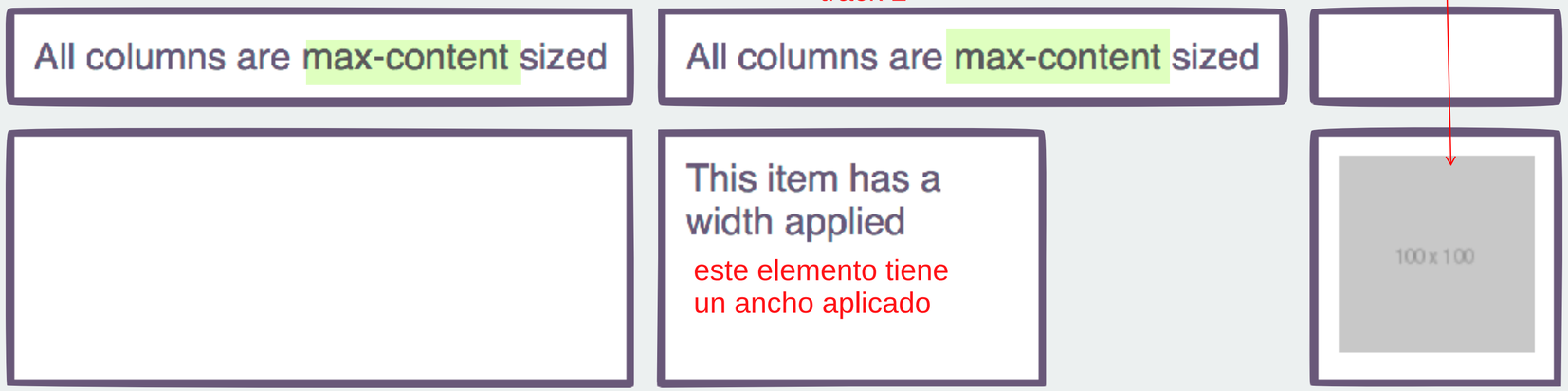
bastante amplio ajustar el texto sin envoltura de texto

Tracks 1 and 2 need to be wide enough to fit the unwrapped text.

A 100px image defines the size of track 3

track 1

track 2



width: 200px;

Intrinsic Sizing

fit-content

Actua como

Act like max-content until it reaches the passed in value.

hasta que alcanza el

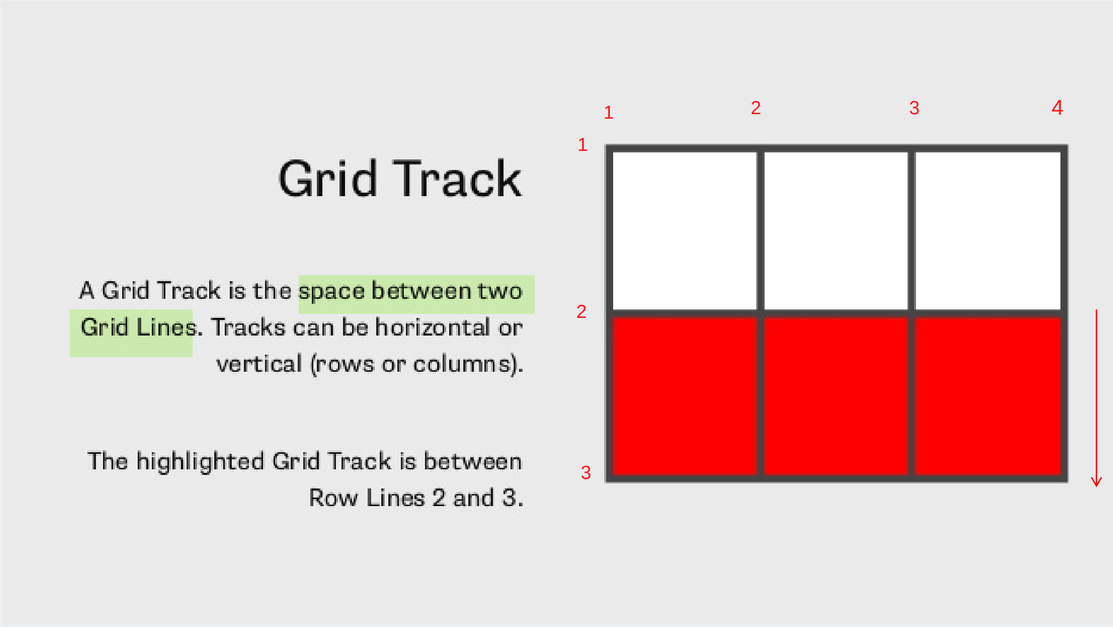
traspasado dentro valor

https://codepen.io/rachelandrew/pen/Mrdobm

Intrinsic Sizing: fit-content

GRID FRANJA/BANDA dimensionado con fit-content
actuara como max-content hasta que ello choque con el limite dado

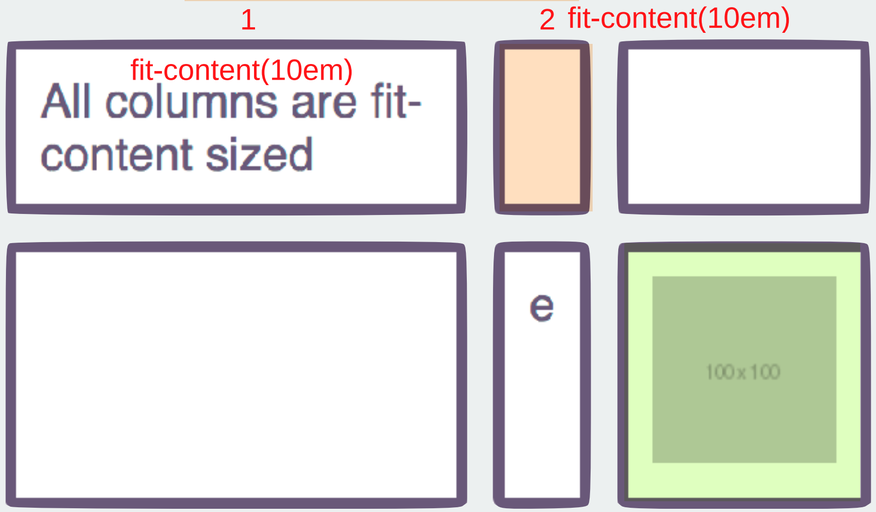
Grid tracks sized with fit-content will act like max-content until they hit the limit given.




```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns:
    fit-content(10em)
    fit-content(10em)
    fit-content(15ch);
}
```

https://codepen.io/rachelandrew/pen/Mrdobm

Columns 1 and 2 are both fit-content(10em). Track 1 wraps at 10em. Track 2 is max-content.



Track 3 is fit-content(15ch)

Flexible lengths

Sizing with fr units

The fr unit describes a flexible length

https://codepen.io/rachelandrew/pen/QaXvPe

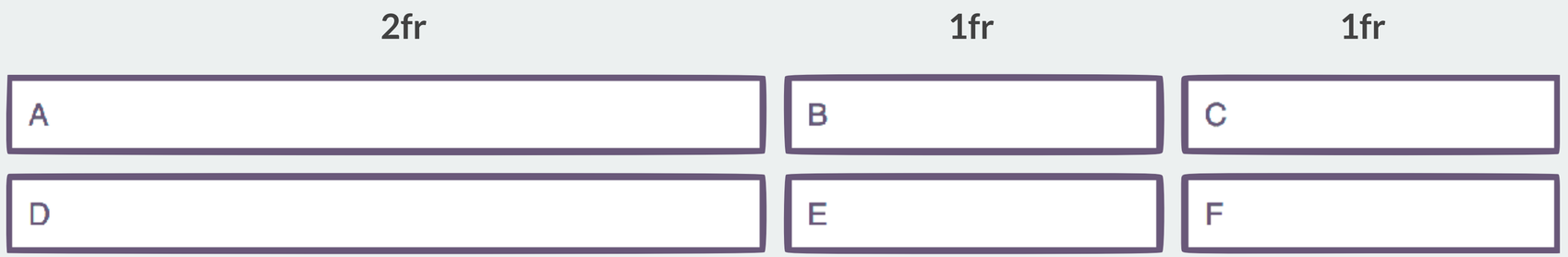
Flexible lengths

The fr unit is a <flex> unit and represents a portion of the available space in the Grid Container.

Porcion

```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 2fr 1fr 1fr;
}
```

https://codepen.io/rachelandrew/pen/QaXvPe



Minimum & Maximum sizing functions

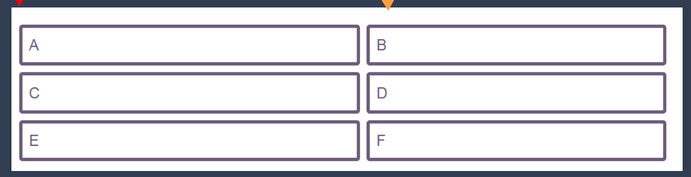
minmax()

funcion minimo tamaño

Minimum Sizing Function

The minimum size for these two tracks is 100px, and 10em.

```
.grid {
  display: grid;
  grid-template-columns:
    minmax(100px, auto)
    minmax(10em, 20em);
}
```



Minimum Sizing Function

banda/franjas

The minimum size of these two tracks is auto.

```
.grid {
  display: grid;
  grid-template-columns:
    10fr
    fit-content(10em);
}
```

1º Tamaño es 10fr del total lo que significa que ocupa todo el ancho dejando una pequeña parte que es ocupado por fit-content(10em)
que al ser fit-content coge solo el valor más pequeño de 10em que sera 1em

Minimum Sizing Function

---

franja/banda

The minimum size of the first track is 100px, the second track is 50px (10% of 500).

```
.grid {
  width: 500px;
  display: grid;
  grid-template-columns:
    100px
    10%; -> 10% de 500px = 50px
}
```

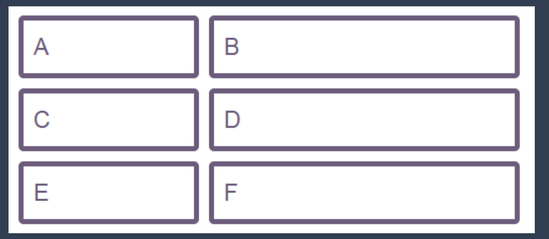
Maximum Sizing Function

The maximum size for these two tracks is 400px, and 20em.

```
.grid {
  display: grid;
  grid-template-columns:
    minmax(100px, 400px)
    minmax(10em, 20em);
}
```

La 2º columna coge ancho a la 1º columna porque tiene mayor anchura

100px a 400px 10em a 20em



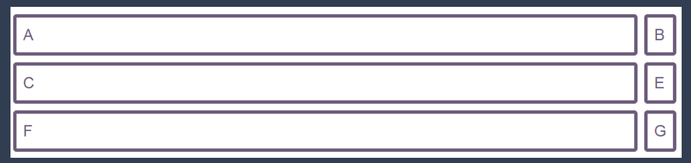
Maximum Sizing Function

primera franja/banda

The maximum size for the first track is max-content.

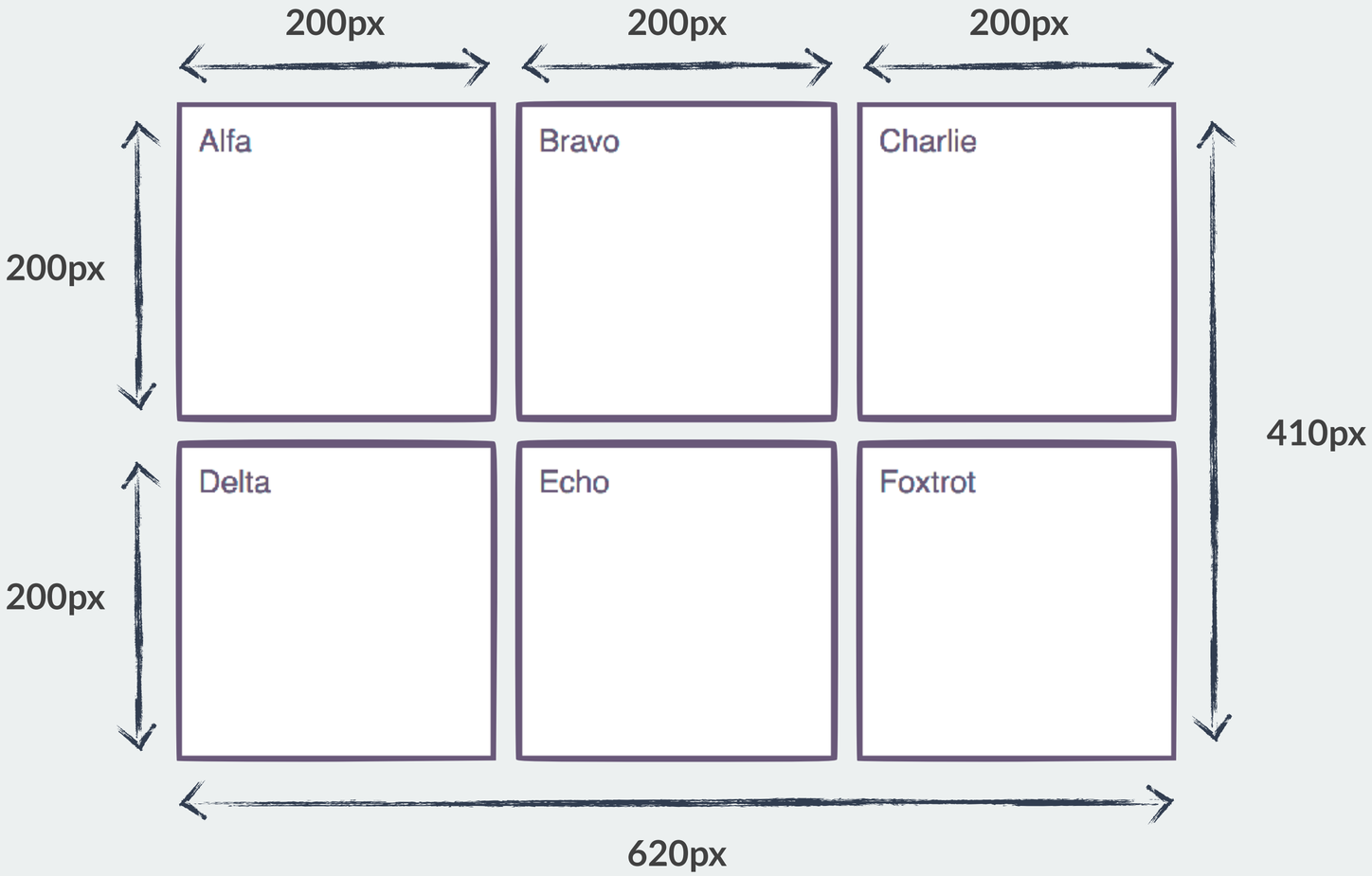
For the second track the maximum is 10em.

```
.grid {
  display: grid;
  grid-template-columns:
    auto maximum size
    fit-content(10em);
}
```



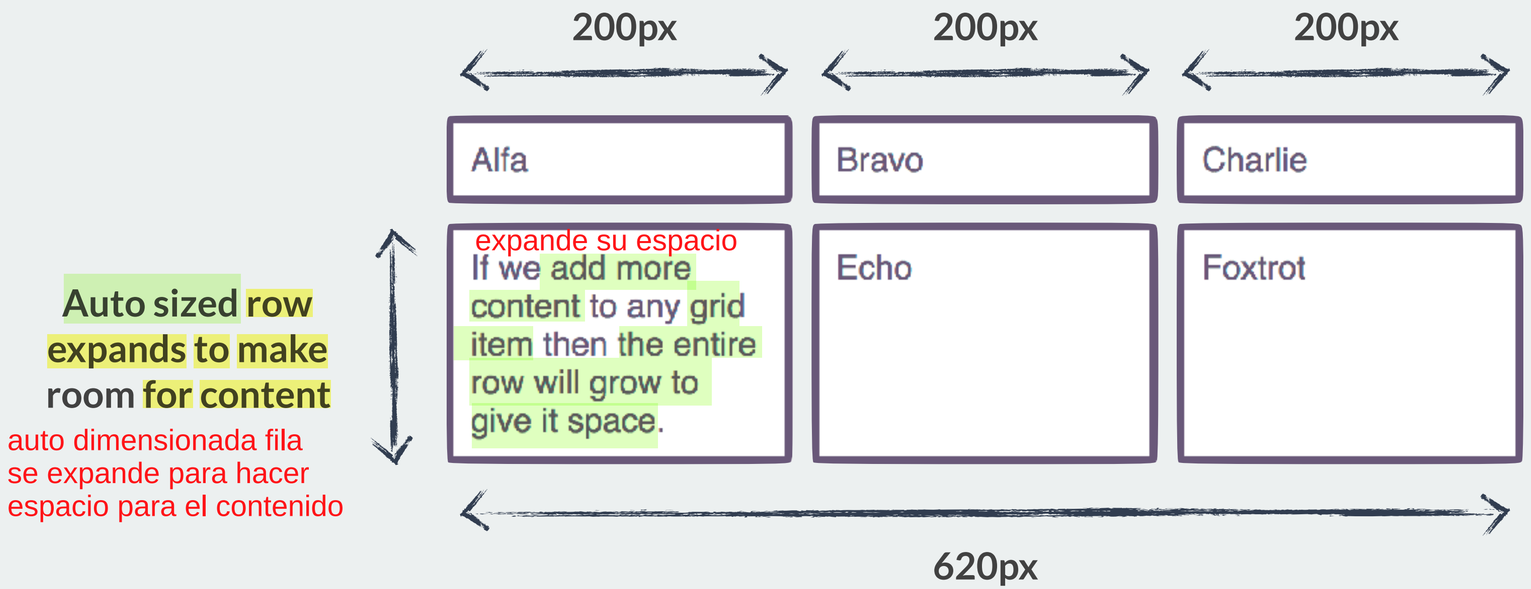
https://codepen.io/rachelandrew/pen/aEgwKY

```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 200px 200px 200px;
  grid-template-rows: 200px 200px;
}
```



```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 200px 200px 200px;
}
```

NO TIENE ASIGNADO NINGUN
TAMAÑO PARA FILAS
No hay
grid-template-rows
grid-auto-rows



```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns: minmax(0,1fr) minmax(0,1fr)  minmax(0,1fr) ;
}
```

La fila auto dimensionada expande para hacer espacio al contenido

Auto sized row expands to make room for content

¿Qué es una unidad de fr de todos modos?

What is a fr unit anyway?

de todos modos

franja/banda

actualmente dimensionado

A track sized with 1fr, is actually sized minmax(auto, 1fr).

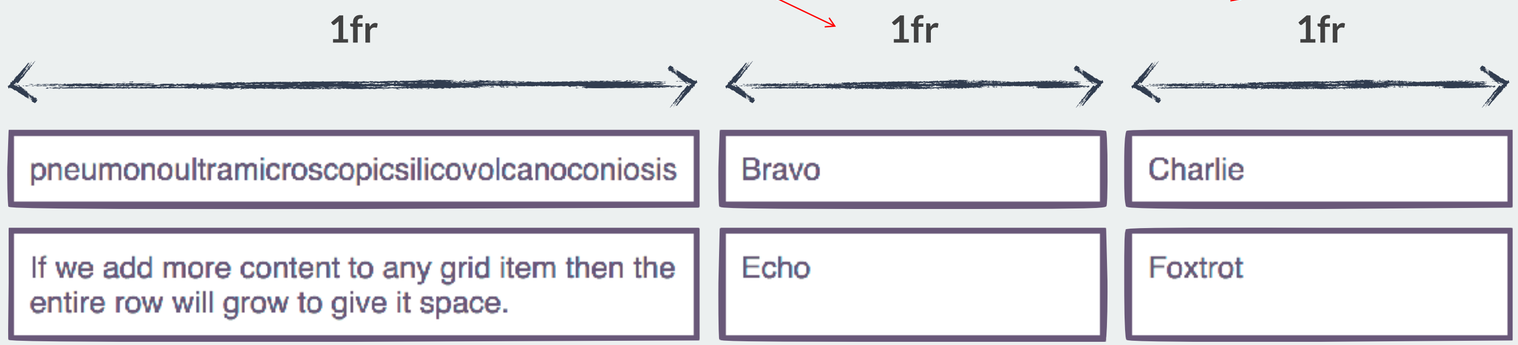
The minimum size is auto.

se auto-ajusta

```
.grid {
  display: grid;
  grid-template-columns: 1fr 1fr 1fr;
}
```

https://codepen.io/rachelandrew/pen/BJgdKL

```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns: minmax(0,1fr) minmax(0,1fr)  minmax(0,1fr) ;
}
```



What is a fr unit anyway?

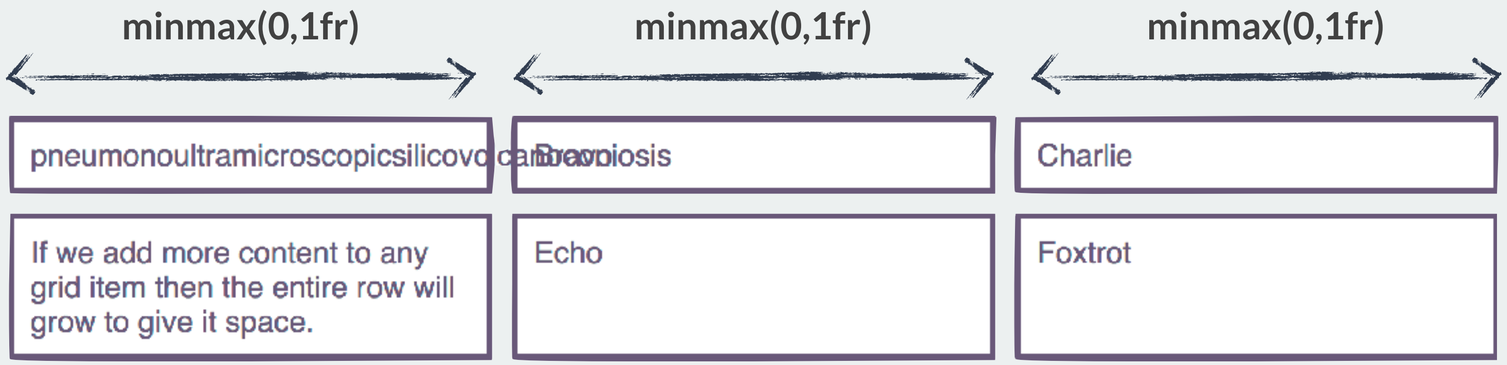
de todos modos

Make the minimum size 0, to force three equal width columns - even if that means overflows.

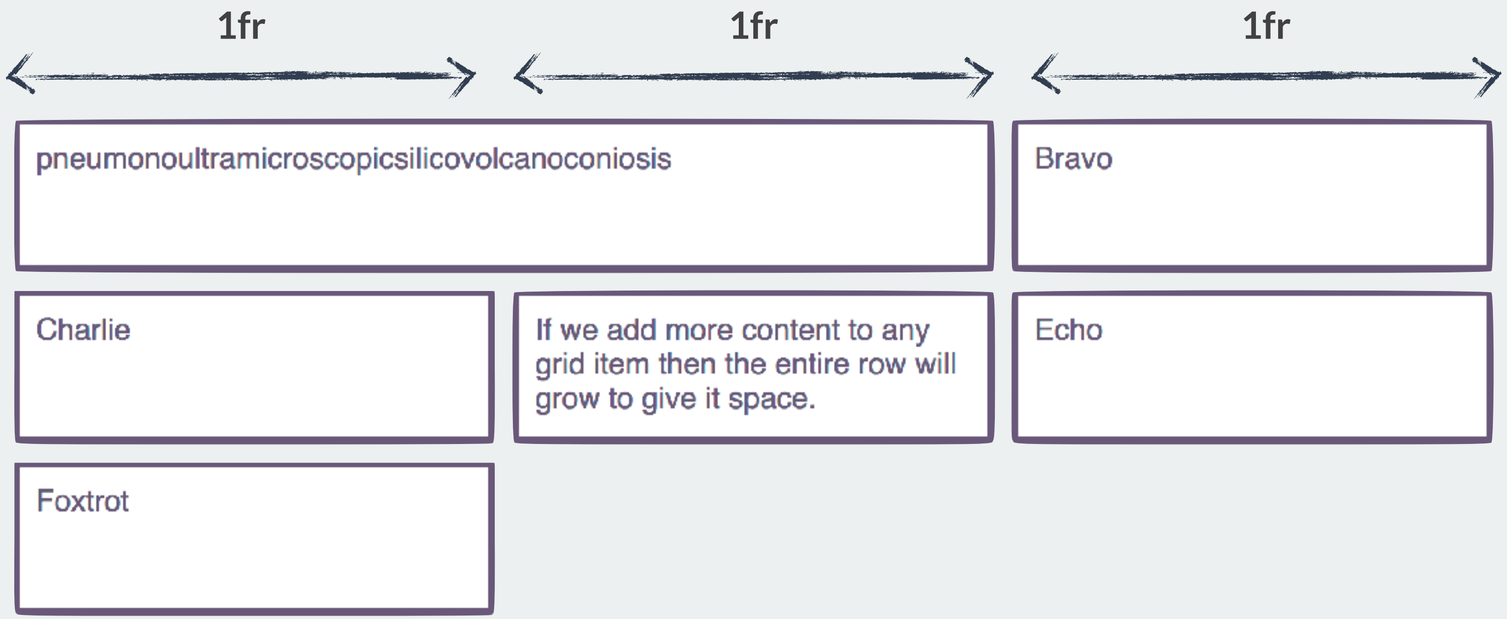
desbordamiento

```
.grid {
  display: grid;
  grid-template-columns:
    minmax(0,1fr)
    minmax(0,1fr)
    minmax(0,1fr);
}
```

3 columnas con desbordamiento



Si agregamos mas contenido
a algun grid item entonces
el total de la fila crecera dar todo el espacio


1fr
1fr
1fr
pneumonoultramicroscopicsilicovolcanoconiosis
Bravo
Charlie
If we add more content to any grid item then the entire row will grow to give it space.
Echo
Foxtrot

https://codepen.io/rachelandrew/pen/WdVjdg

Tantas columnas flexibles como quepan

As many flexible columns as will fit

---

ajusta estira a lo largo

200px

Use auto-fill or auto-fit along with repeat and minmax, to create as many flexible columns with a minimum size as will fit into the container.

crear como varias columnas flexibles con un mínimo tamaño que se ajustara dentro del contenedor

```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns: repeat(auto-fill, minmax(200px, 1fr));
  grid-auto-rows: 100px;
}

.grid :nth-child(odd) {
  grid-row: auto / span 2;
}

.grid :nth-child(3n) {
  grid-column: auto / span 2;
}

.grid > * {
  border: 5px solid rgb(108,91,123);
  border-radius: 5px;
  padding: 10px;
}
```

CUANDO LLEGAMOS AL MINIMO TAMAÑO DEL NAVEGADOR
LOS GRID-ITEMS mas pequeños no pasan de 200px

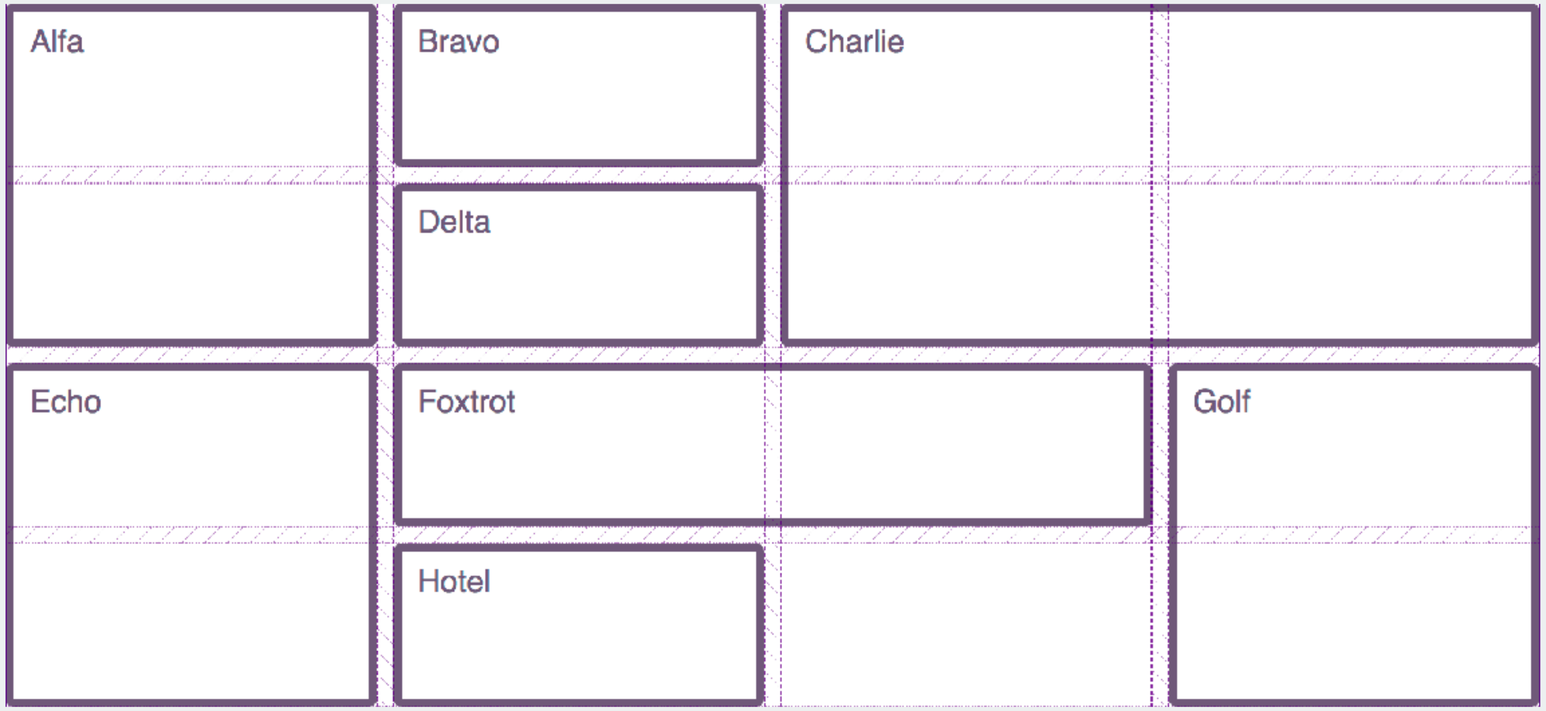
```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns:
    repeat(auto-fill, minmax(200px, 1fr));
  grid-auto-rows: 100px;
}
```

auto-fill

100px

100px

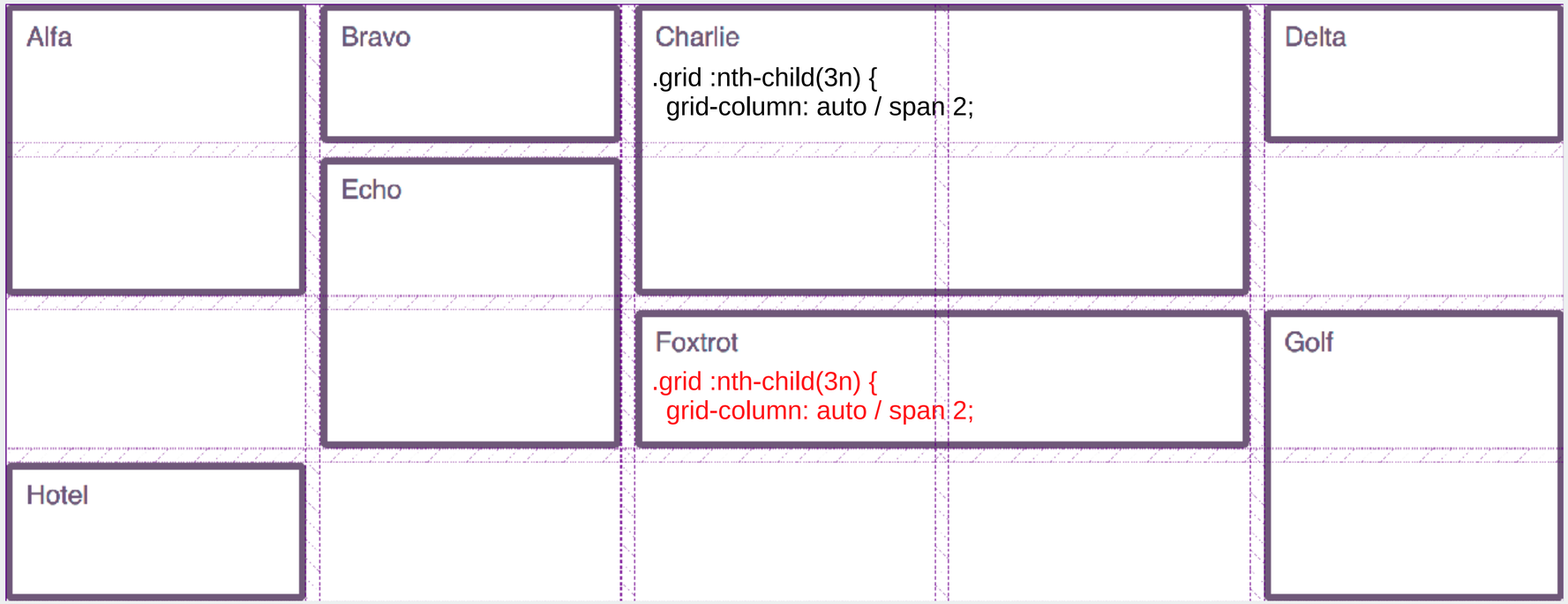
https://codepen.io/rachelandrew/pen/WdVjdg



```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns: repeat(auto-fill, minmax(200px, 1fr));
  grid-auto-rows: 100px;
}

.grid :nth-child(odd) {
  grid-row: auto / span 2;
}

.grid :nth-child(3n) {
  grid-column: auto / span 2;
}
```



https://codepen.io/rachelandrew/pen/ZvgKNY

Dense Packing Mode

Using the value dense for grid-auto-flow will cause items to be displayed out of document order.

```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns: repeat(auto-fill, minmax(200px, 1fr));
  grid-auto-rows: 100px;
  grid-auto-flow: dense;
}

.grid :nth-child(odd) {
  grid-row: auto / span 2;
}

.grid :nth-child(3n) {
  grid-column: auto / span 2;
}

.grid > * {
  border: 5px solid rgb(108,91,123);
  border-radius: 5px;
  padding: 10px;
}
```

```
.grid {
  display: grid;
  grid-gap: 10px;
  grid-template-columns:
    repeat(auto-fill, minmax(200px, 1fr));
  grid-auto-rows: 100px;
  grid-auto-flow: dense;
}
```

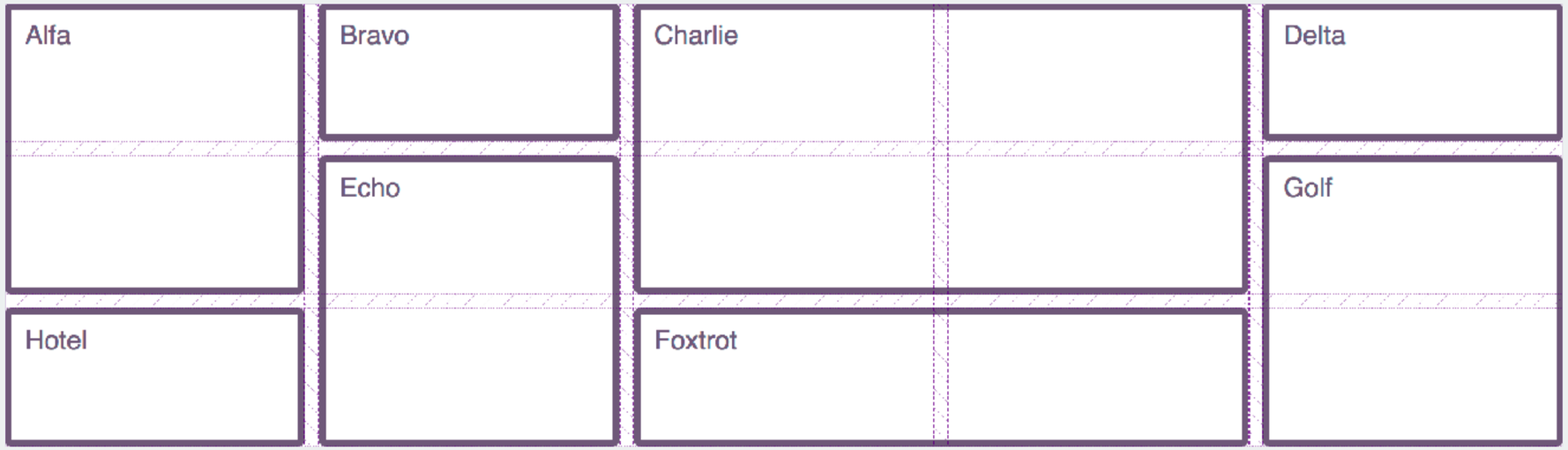
https://codepen.io/rachelandrew/pen/ZvgKNY

Adaptable 1294 × 1080

repeat(auto-fill, minmax(200px, 1fr));

Anchura minima : 200px
Anchura maxima : 1fr

auto-fill



This must be pretty slow, **right**?

CSS performance test: Flexbox v CSS Table – Fight!

26.03.2014 7 comments

1402 DAYS 1402 days since this post was last revised. Specific details are likely out of date.

Is the latest Flexbox implementation slower than CSS table layout? Yes, it is. But not enough to concern yourself about. If you need to use Flexbox, use it. For all but the most extreme situations, any speed difference against other layout methods is largely inconsequential.

If you want to know why you would employ `display: table` and `display: table-cell` in the first place, I wrote a post in July 2013 covering the benefits of using these properties for responsive web design: http://benfra.in/20m

Introduction
Methodology
Measuring performance
Chrome Version 33.0.1750.152 (Mac)
Firefox v28 (Mac)
Opera 20 (Mac)
Internet Explorer 11.01
Chrome Version 33.0.1750.152 (Android 4.3)
Firefox 26.0.1 (Android 4.3)
Comparison testing at WebPageTest
Results summary
Conclusion

Introduction

I'm a flex layout fan. I was writing primers on it back in April 2012, so don't assume I'm some Flex layout hater.

I would like to make it clear that the only relevant thing in these figures is the difference. The technique employed is measuring many things beside layout alone (style calculation, rendering etc). I also have concerns at this point about the small amount of 'runs' used in this test (aka what statisticians refer to as the danger of small numbers). In the future I hope to write something that performs an infinitely greater number of runs and presents them without the need to manually refresh.

A Flexbox Grid with Bootstrap 4

Uses flexbox for the grid, I used the default Bootstrap grid classes.

Usa flexbox para la cuadrícula, yo
usé las clases de cuadrícula por defecto de Bootstrap.

```
<div class="container-fluid">
  <div class="row">
    <div class="col-sm">One of three columns</div>
    <div class="col-sm">One of three columns</div>
    <div class="col-sm">One of three columns</div>
  </div>
  <div class="row">
    <div class="col-sm">One of four columns</div>
    <div class="col-sm">One of four columns</div>
    <div class="col-sm">One of four columns</div>
    <div class="col-sm">One of four columns</div>
  </div>
  <div class="row">
    <div class="col">1 of 3</div>
    <div class="col-6">2 of 3 (wider)</div>
    <div class="col">3 of 3</div>
  </div>

  <!— repeat 999 times —>
</div>
```

A Float Based Grid with Column Setter

Using floats with this Sass-based framework.

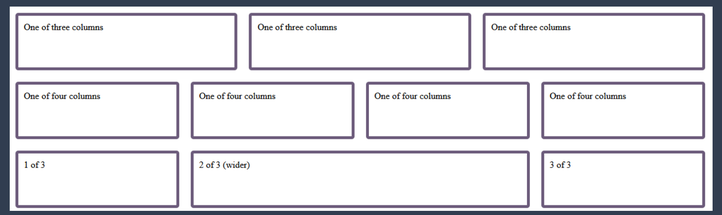
```
.grid {
    max-width: 1200px;
    margin: 40px auto;
    display: grid;
    grid-template-columns: repeat(12, 1fr);
    grid-auto-rows: 100px;
    grid-gap: 20px;
}

.grid>div {
    border: 5px solid rgb(108, 91, 123);
    border-radius: 5px;
    padding: 10px;
}

.four-cols {
    grid-column: auto / span 4;
}

.three-cols {
    grid-column: auto / span 3;
}

.six-cols {
    grid-column: auto / span 6;
}
```




```
.row::after {
  content: "";
  display: block;
  clear: both; }

.row > div {
  border: 5px solid #6c5b7b;
  border-radius: 5px;
  padding: 10px;
  min-height: 100px;
  margin-bottom: 20px; }

.four-cols {
  width: 31.42857%;
  float: left;
  margin-right: 2.85714%; }

.three-cols {
  width: 22.85714%;
  float: left;
  margin-right: 2.85714%; }

.six-cols {
  width: 48.57143%;
  float: left;
  margin-right: 2.85714%; }

.row > :last-child {
  margin-right: 0;
}
```

CSS Grid

envoltura de las filas

We lost the row wrappers in the HTML for the Grid version.

```
.grid {
  display: grid;
  grid-template-columns: repeat(12, 1fr);
  grid-auto-rows: 100px;
  grid-gap: 20px;
}

.four-cols { grid-column: auto / span 4; }

.three-cols { grid-column: auto / span 3; }

.six-cols { grid-column: auto / span 6; }
```

1fr 1fr 1fr

One of three columns | One of three columns | One of three columns

One of four columns | One of four columns | One of four columns | One of four columns

1 of 3 | 2 of 3 (wider) | 3 of 3

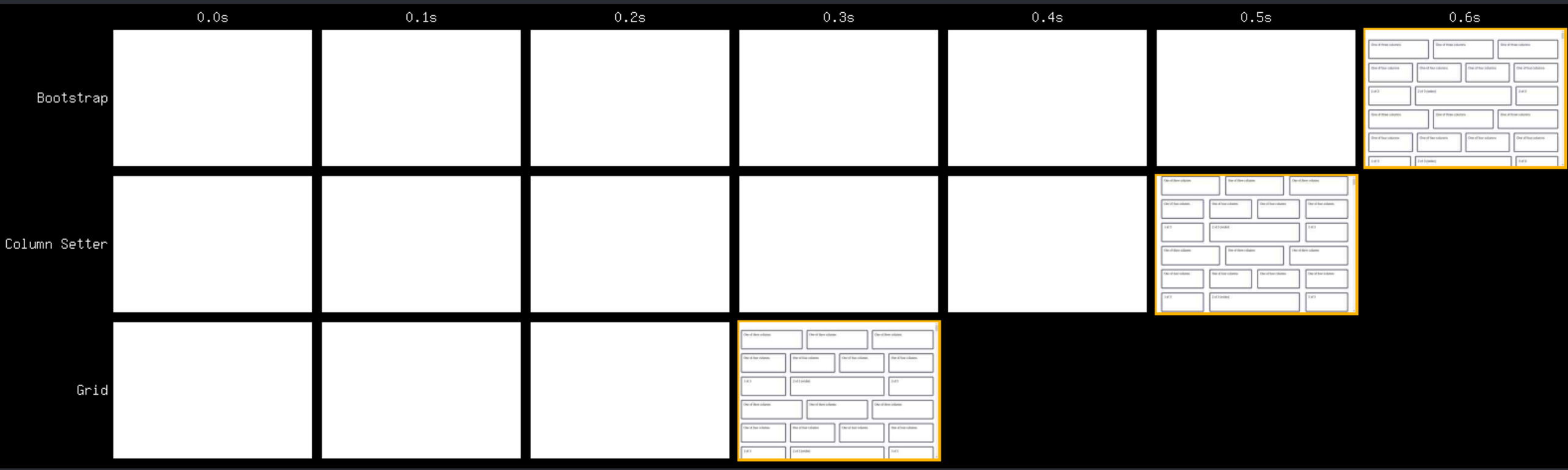
One of three columns | One of three columns | One of three columns

One of four columns | One of four columns | One of four columns | One of four columns

1 of 3 | 2 of 3 (wider) | 3 of 3

One of three columns | One of three columns | One of three columns

One of four columns | One of four columns | One of four columns | One of four columns


0.0s
0.1s
0.2s
0.3s
0.4s
0.5s
0.6s
Bootstrap
Column Setter
Grid

	0.0s	0.1s	0.2s	0.3s	0.4s
fr units					
auto					
fixed length					
min-content					

“Where is my magic Grid Polyfill?”

–everyone on the web

https://www.reddit.com/r/web_design/comments/700tfh/is_there_a_working_polyfill_for_css_grid_for_ie_11/

http://brm.io/jquery-match-height-demo/

jquery.matchHeight Browser Tests

Project page · GitHub · 11 tests passed, 0 failed

padding margin border border-box by row remove matchHeight show hidden

Lorem ipsum

Phasellus ut nibh fermentum, vulputate urna vel, semper diam.

Aenean semper felis ipsum, vulputate consequat dui elementum vel.

Lorem ipsum dolor

Phasellus ut nibh fermentum, vulputate urna vel, semper diam. Nunc sollicitudin felis ut pellentesque fermentum. In erat mi, pulvinar sit amet tincidunt vitae, gravida id felis. Phasellus hendrerit erat sed porta imperdiet. Vivamus viverra ipsum tortor, et congue mauris porttitor ut.

Test content update.

Lorem ipsum dolor sit amet.

Aenean semper felis ipsum, vulputate consequat dui elementum vel. Nullam odio eros, sagittis vitae lectus id, pretium viverra lectus. Etiam auctor dolor non dui ultricies pulvinar.

Lorem ipsum dolor

Aenean semper.

Lorem ipsum dolor

Phasellus ut nibh fermentum, vulputate urna vel, semper diam. Nunc sollicitudin felis ut pellentesque fermentum. In erat mi, pulvinar sit amet tincidunt vitae, gravida id felis.

Lorem ipsum dolor

Aenean semper felis ipsum, vulputate consequat dui elementum vel.

Lorem ipsum dolor

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Lorem ipsum dolor

Phasellus ut nibh fermentum, vulputate urna vel, semper diam.

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.

One of three columns

One of three columns

One of four columns

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. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke.

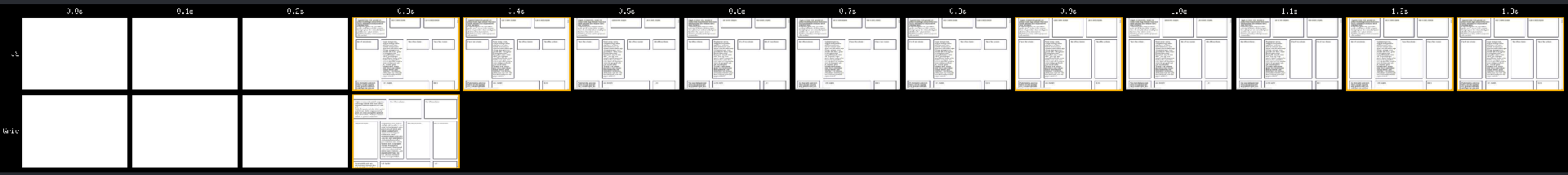
One of four columns

One of four columns

Pea horseradish azuki bean lettuce avocado asparagus okra. Kohlrabi radish okra azuki bean corn fava bean mustard tigernut jícama green bean celtuce collard greens avocado quandong fennel gumbo black-eyed pea. Grape silver beet watercress potato tigernut corn groundnut. Chickweed okra pea winter purslane coriander yarrow sweet pepper radish garlic brussels sprout groundnut summer purslane earthnut pea tomato spring onion azuki bean gourd.

2 of 3 (wider)

3 of 3


0.0s
0.1s
0.2s
0.3s
0.4s
0.5s
0.6s
0.7s
0.8s
0.9s
1.0s
1.1s
1.2s
1.3s

Sorry.

We develop in an environment far more powerful than that of our users.

Testing often does not reflect realistic day to day use, or realistic data.

“Where is my magic Grid Polyfill?”

please stop asking this question

https://caniuse.com/#search=grid

CSS Grid Layout - CR

Method of using a grid concept to lay out content, providing a mechanism for authors to divide available space for layout into columns and rows using a set of predictable sizing behaviors. Includes support for all `grid-*` properties and the `fr` unit.

Usage % of all users

Global 72.4% + 4.31% = 76.71%

unprefixed: 72.4%

Current aligned | Usage relative | Date relative | Show all

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Opera Mobile	Chrome for Android	Firefox for Android	IE Mobile	UC Browser for Android	Samsung Internet	QQ Browser	Baidu Browser
			1 49			9.3										
			61			10.2										
	2 15	57	62	10.1	49	10.3		4.4						4		
2 11	16	58	63	11	50	11.2	all	62	37	64	57	2 11	11.4	6.2	1 1.2	7.12
	17	59	64	11.1	51	11.3										
		60	65	TP	52											
		61	66													
			67													

Notes | Known issues (2) | Resources (13) | Feedback

1 Enabled in Chrome through the "experimental Web Platform features" flag in chrome://flags

2 Partial support in IE refers to supporting an older version of the specification.

https://caniuse.com/#search=grid

CSS Grid Layout - CR

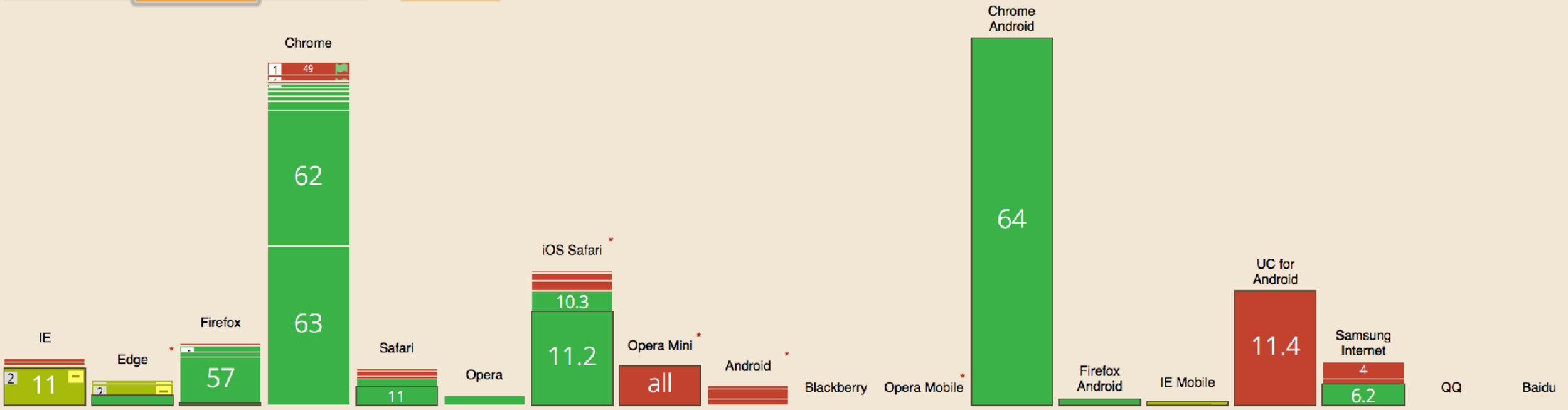
Method of using a grid concept to lay out content, providing a mechanism for authors to divide available space for layout into columns and rows using a set of predictable sizing behaviors. Includes support for all `grid-*` properties and the `fr` unit.

Usage % of all users

Global 72.4% + 4.31% = 76.71%

unprefixed: 72.4%

Current aligned | Usage relative | Date relative | Showing all



Notes | Known issues (2) | Resources (13) | Feedback

1 Enabled in Chrome through the "experimental Web Platform features" flag in chrome://flags

New CSS can help you create **better experiences** in browsers that don’t support new CSS.

A 12 column layout

```
.grid {
  display: grid;
  grid-template-columns: repeat(12, 1fr);
  grid-auto-rows: 100px;
  grid-gap: 20px;
}

.four-cols { grid-column: auto / span 4; }

.three-cols { grid-column: auto / span 3; }

.six-cols { grid-column: auto / span 6; }
```

New CSS knows about old CSS

- **Media Queries**:
  tailor your design according to features of the device
- **Feature Queries**:
  tailor your design to the support in the browser

https://codepen.io/rachelandrew/pen/aqbdOy

Heading here

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.

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

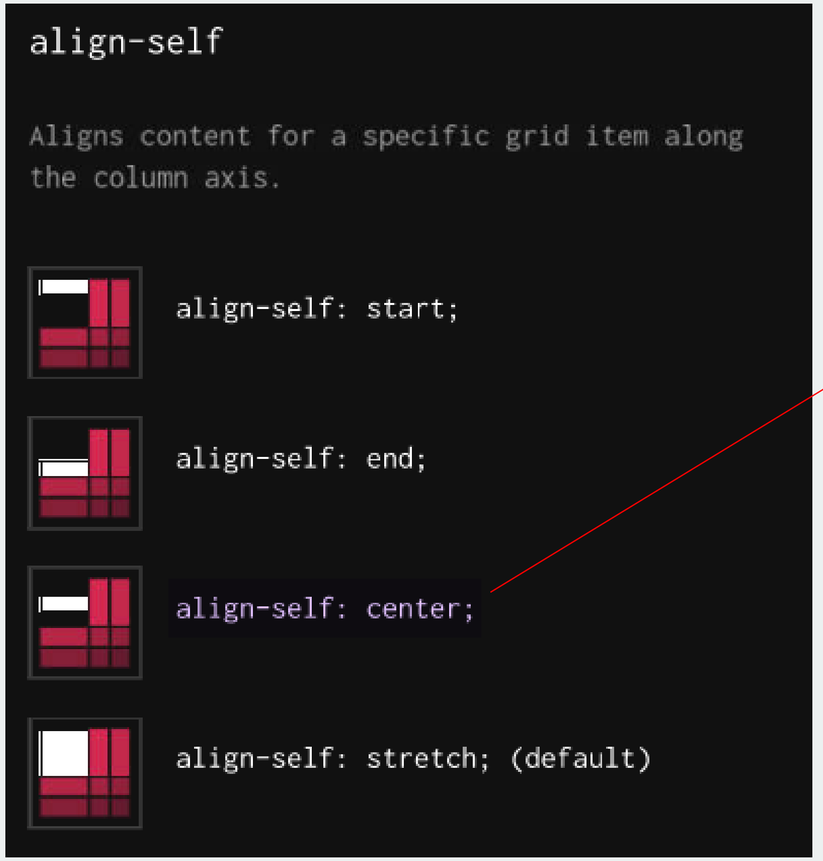
https://codepen.io/rachelandrew/pen/aqbdOy

añadimos una linea centro vertical o bien lado del texto

Adding a vertical centre line either side of the text

Usando generado contenido añadir un borde

Using generated content to add a border.




```
header h1 {
  text-align: center;
  display: grid;
  grid-template-columns: 1fr auto 1fr;
  grid-gap: 20px;
}

header h1:before,
header h1:after {
  content: "";
  align-self: center;
  border-top: 1px solid rgba(37,46,63,.5);
}
```

https://codepen.io/rachelandrew/pen/aqbdOy

Heading here

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.

Veggies es bonus vobis, proinde vos postulo essum magis kohlrabi welsh onion daikon amaranth tatsoi tomatillo melon azuki bean garlic.

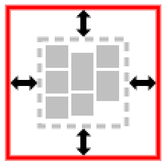
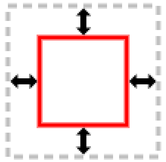
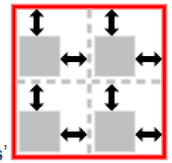
Gumbo beet greens corn soko endive gumbo gourd. Parsley shallot courgette tatsoi pea sprouts fava bean collard greens dandelion okra wakame tomato. Dandelion cucumber earthnut pea peanut soko zucchini.

Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water

https://codepen.io/rachelandrew/pen/aqbdOy

Feature Queries

Test for a CSS feature, and apply the CSS if it exists in that browser.

Common	Axis	Aligns	Applies to
'justify-content'	main/inline	content within element (effectively adjusts padding) 	block containers, flex containers, and grid containers
'align-content'	cross/block		
'justify-self'	inline	element within parent (effectively adjusts margins) 	block-level boxes, absolutely-positioned boxes, and grid items
'align-self'	cross/block		absolutely-positioned boxes, flex items, and grid items
'justify-items'	inline	items inside box (controls child items' 'justify-self: auto') 	block containers and grid containers
'align-items'	cross/block		flex containers and grid containers

```
header h1 {
  text-align: center;
  display: grid;
  grid-template-columns: 1fr auto 1fr;
  grid-gap: 20px;
}

@supports (display: grid) {
  header h1:before,
  header h1:after {
    content: "";
    align-self: center;
    border-top: 1px solid rgba(37,46,63,.5);
  }
}
```

https://codepen.io/rachelandrew/pen/aqbdOy

No Grid

Heading here

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.

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

Grid

Heading here

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.

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

Creating circles from squares

Use border-radius

Don't forget that old CSS still exists!

```
header img {
  border-radius: 50%;
  margin: 0 auto 2em auto;
}
```

https://codepen.io/rachelandrew/pen/aqbdOy

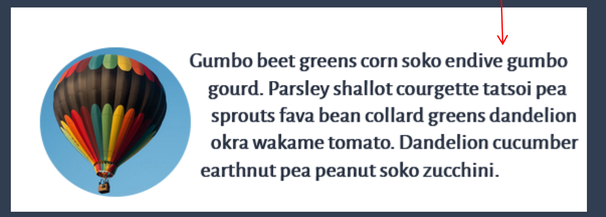
CSS Shapes

flotando el texto y curvando el texto alrededor
una vez la pantalla esta ampliada bastante hacer esto

Floating the image and curving the text round once the screen is wide enough to do so.

```
header img {
  border-radius: 50%;
  margin: 0 auto 2em auto;
}

                        480px
@media (min-width: 30em) {
  header img {
    float: left;
    shape-outside: margin-box;
    margin: 0 20px 0 0;
  }
}
```



https://codepen.io/rachelandrew/pen/aqbdOy

Heading here

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.

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. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke.

https://codepen.io/rachelandrew/pen/aqbdOy

Heading here

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.

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. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean

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.

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. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke.

Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

https://codepen.io/rachelandrew/pen/aqbdOy

Multi-column layout

Well supported, responsive by default.

```
section {
  column-width: 15em;
}
```

240px

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. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke.

Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

https://codepen.io/rachelandrew/pen/aqbdOy

Vertical Media Queries

tenemos suficiente altura mostar las columnas sin scroll vertical

Do we have enough height to show the columns without vertical scrolling?

Si hay más de 500px de altura

```
@media (min-height: 500px) {
  section {
    column-width: 15em;
  }
}
```

240px

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. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke.

Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

https://codepen.io/rachelandrew/pen/aqbdOy

Heading here

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.

120px

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

120px

green bean swiss chard seakale pumpkin onion chickpea gram corn pea. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke.

Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

https://codepen.io/rachelandrew/pen/aqbdOy

Creating columns of cards

Coincidiendo el minimo ancho grid

y el espacio del ancho de la multi columna

Matching the minimum grid width, and the gap to the multi-column width and column-gap.

y espacio columna

```
.resources {
  max-width: 60em;
}

.resources {
  margin: 1em auto;
  padding: 0;
  list-style: none;
  display: grid;
  grid-template-columns:
    repeat(auto-fill, minmax(15em,1fr));
  grid-gap: 1em;

}

.resources li.image {
  grid-row: auto / span 2;
}
```

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 fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

Title

Pea horseradish azuki bean lettuce avocado asparagus okra. Kohlrabi radish okra azuki bean corn fava bean mustard tigernut jícama green bean celtuce collard greens avocado quandong fennel gumbo black-eyed pea.

Title

Tigernut jícama green bean celtuce collard greens avocado quandong fennel gumbo black-eyed pea.

Title

Pea horseradish azuki bean lettuce avocado asparagus okra. Kohlrabi radish okra azuki bean corn fava bean mustard tigernut jícama green bean celtuce collard greens avocado quandong fennel gumbo black-eyed pea.

https://codepen.io/rachelandrew/pen/aqbdOy

retirada

Using in-line: block as a fallback

---

anular

We only need to over-ride the width and margin in a Feature Query.

Si no soporta CSS GRID LAYOUT 640px

```
@media (min-width: 40em) {
    .resources li {
      display: inline-block;
      width: 47%;
      margin: 0 1% 1em 1%;
      vertical-align: top;
  }

}
```

Soporta CSS GRID LAYOUT

```
@supports (display: grid) {
  .resources li {
    width: auto;
    margin: 0;
  }
}
```

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 fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

Title

Pea horseradish azuki bean lettuce avocado asparagus okra. Kohlrabi radish okra azuki bean corn fava bean mustard tigernut jícama green bean celtuce collard greens avocado quandong fennel gumbo black-eyed pea.

Title

Pea horseradish azuki bean lettuce avocado asparagus okra. Kohlrabi radish okra azuki bean corn fava bean mustard tigernut jícama green bean celtuce collard greens avocado quandong fennel gumbo black-eyed pea.

Title

Title

retroceder

Using multi-column layout as a fallback

No necesitamos anular las propiedades de las columnas

We don't need to override the column-* properties. They are ignored once we are in grid layout.

Son ignoradas una vez que nosotros estamos dentro Grid Layout

640px

```
@media (min-width: 40em) {
  .resources {
    column-width: 15em;
  }

  .resources li {
    break-inside: avoid;
  }
}
```

establece el comportamiento de las columnas

Evita que se inserte cualquier salto (página, columna o región) dentro del cuadro principal.

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 greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

Title

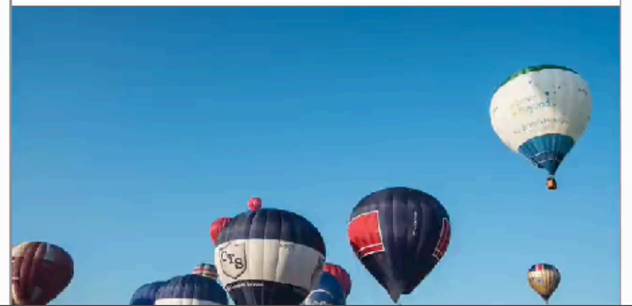
Pea horseradish azuki bean lettuce avocado asparagus okra. Kohlrabi radish okra azuki bean corn fava bean mustard tigernut jícama green bean celtuce collard greens avocado quandong fennel gumbo black-eyed pea.

Title

Pea horseradish azuki bean lettuce avocado asparagus okra. Kohlrabi radish okra azuki bean corn fava bean mustard tigernut jícama green bean celtuce collard greens avocado quandong fennel gumbo black-eyed pea.

Title

Title



Nosotros podemos LOGRAR un monton con un poco CSS

We can achieve a lot with a little CSS.

Antes de preocuparse por si una técnica funciona

Before worrying if a technique performs well, ask yourself if you need it at all.

Bueno, pregúntese si lo necesita.

Replantear buscador soporta conversacion

Reframe the browser support conversation.

We have the tools to provide great experiences for everyone.

https://rachelandrew.co.uk/speaking/event/smashing-london-2018

Thank you!

@rachelandrew
https://rachelandrew.co.uk