

Introducción y primeros pasos con CSS Grid layout

Juan Carlos Ruiz Carrión

@klaufel

Content-2

Content-3

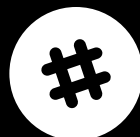
footer



Juan Carlos Ruiz Carrión

UI / Front-end developer

Dev:



@klaufel

Party hard:



@klaufel



Klaufel#2373



Fer



Edu



Juan Carlos

Introducción



“ CSS Grid no solo trata de añadir márgenes, espacios y propiedades a cada elemento y colocarlos, ahora puedes tener un modelo diferente, un modelo en el que puedes desarrollar primero tus elementos y después los puedes insertar en tu diseño ”

Bert Bos

Co-autor de la primera especificación de CSS junto con Håkon Wium Lie.

La historia de CSS Grid a través de sus creadores

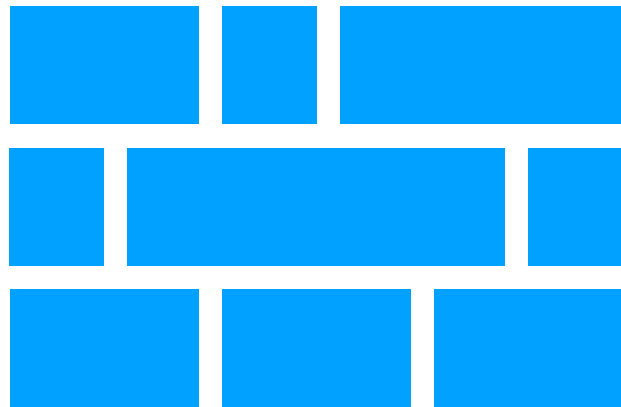
<http://alistapart.com/article/the-story-of-css-grid-from-its-creators>

¿Qué es CSS Grid layout?

- Un nuevo módulo de posicionamiento añadido a la propiedad `display`
- Es una nueva forma de maquetar y colocar la información en 2 dimensiones (x, y)
- Nos permite crear rejillas descritas en CSS otorgándonos la posibilidad de reordenarlas visualmente, independientemente del emplazamiento del elemento en el documento

Grid layout no sustituye a Flexbox, lo complementa

Grid layout



Flexbox



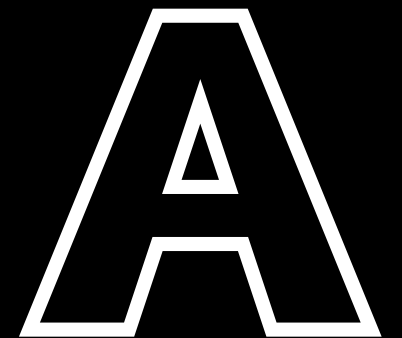
Soporte en navegadores

IE	Edge *	Firefox	Chrome	Safari	iOS Safari *	Opera Mini *	Chrome Android	UC for Android	Samsung Internet
			¹ 49		9.3				
			61		10.2				
	² 15		62	10.1	10.3				4
² 11	16	57	63	11	11.2	all	62	11.4	6.2
	17	58	64	TP					
		59	65						
		60	66						

Global: 72.40% + 4.31% = 76.71%
 España: 89.44% + 2.22% = 91.67%

Partial Support Prefixed

Terminología



Grid container

El elemento al que le declaremos la propiedad "**display: grid**" funcionará como el contenedor de nuestro grid.

Es el padre directo de todos los elementos del grid.

```
<div class="wrapper">
```

```
  . . .
```

```
</div>
```



Grid container

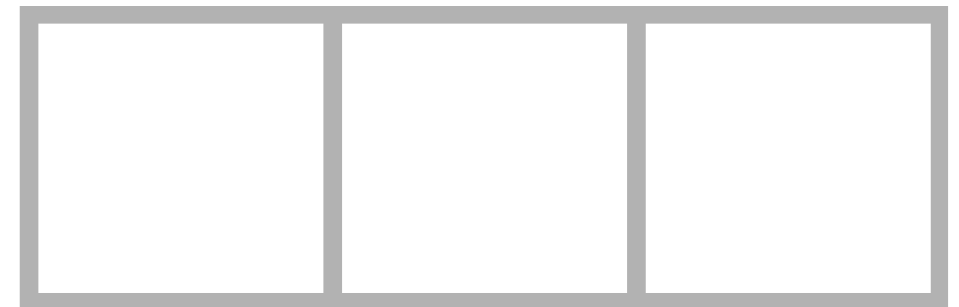
En este ejemplo, ".wrapper" es el grid container

Grid item

Los elementos descendientes (directos) de “grid container”.

La relación entre el elemento padre (grid container) y el hijo (grid item) se denomina “**grid context**”

```
<div class="wrapper">  
  <div class="wrapper__item"></div>  
  <div class="wrapper__item"></div>  
  <div class="wrapper__item"></div>  
</div>
```

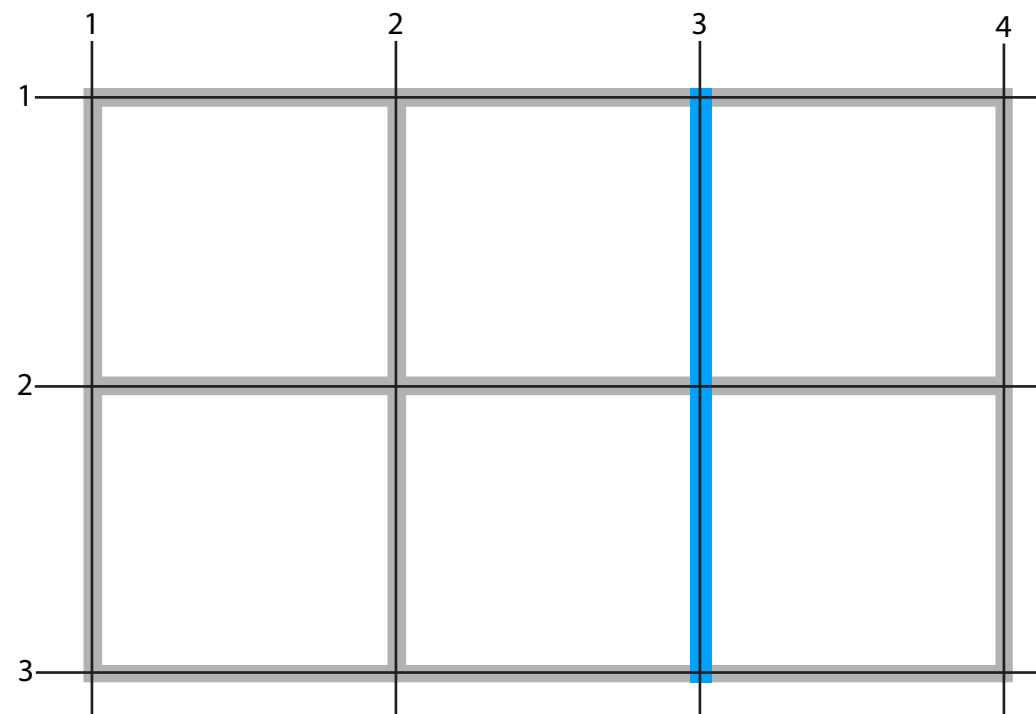


En este ejemplo, “.wrapper__item” son los grid item

Grid line

Las líneas verticales y horizontales que dividen el grid y separan las columnas y las filas.

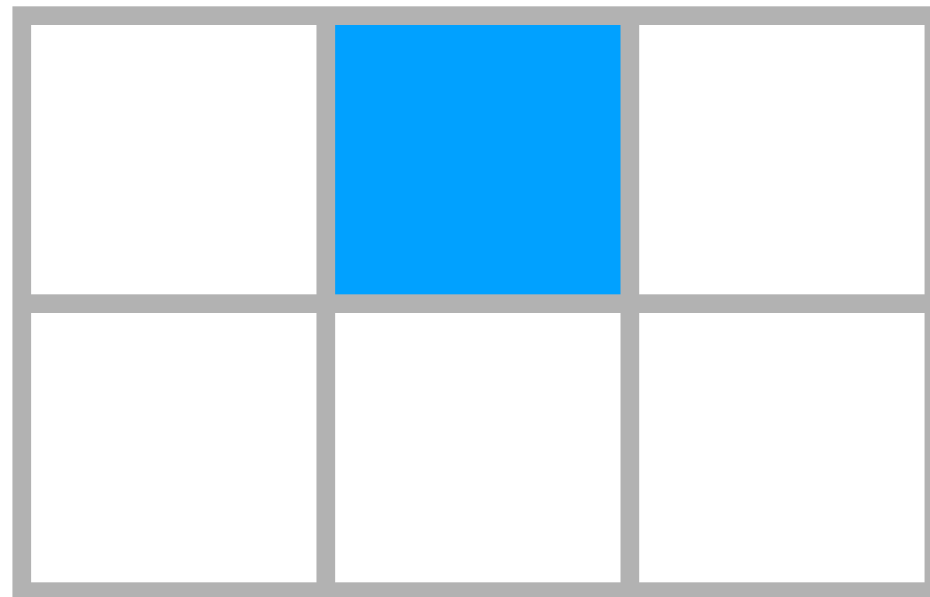
Podemos hacer referencia a una “**grid line**” por número o por nombre.



La línea azul representa una grid line de una columna

Grid cell

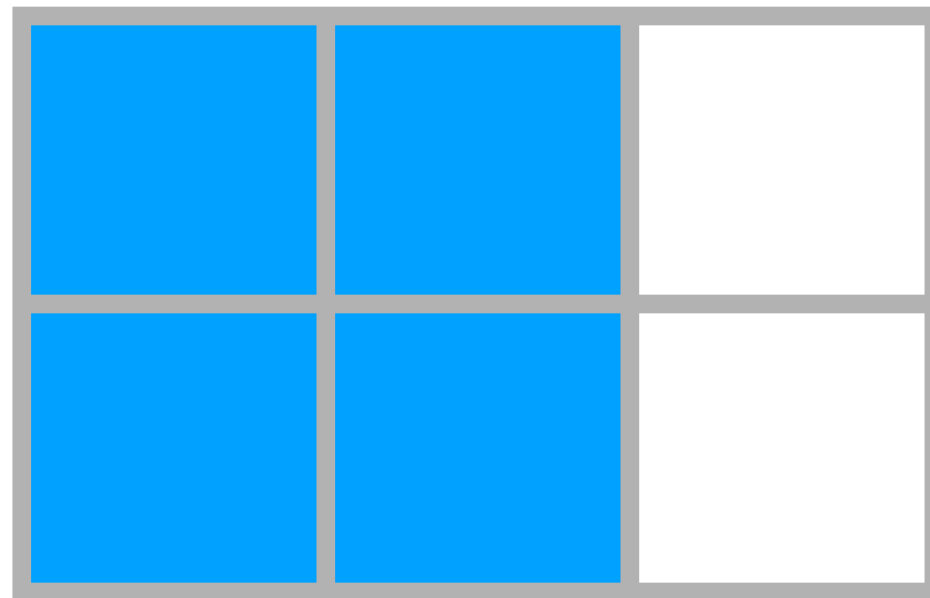
Es una unidad de nuestro grid.



Grid area

Espacio rectangular rodeado por cuatro **grid lines**.

Un **grid area** puede estar compuesta por cualquier cantidad de **grid cells**.

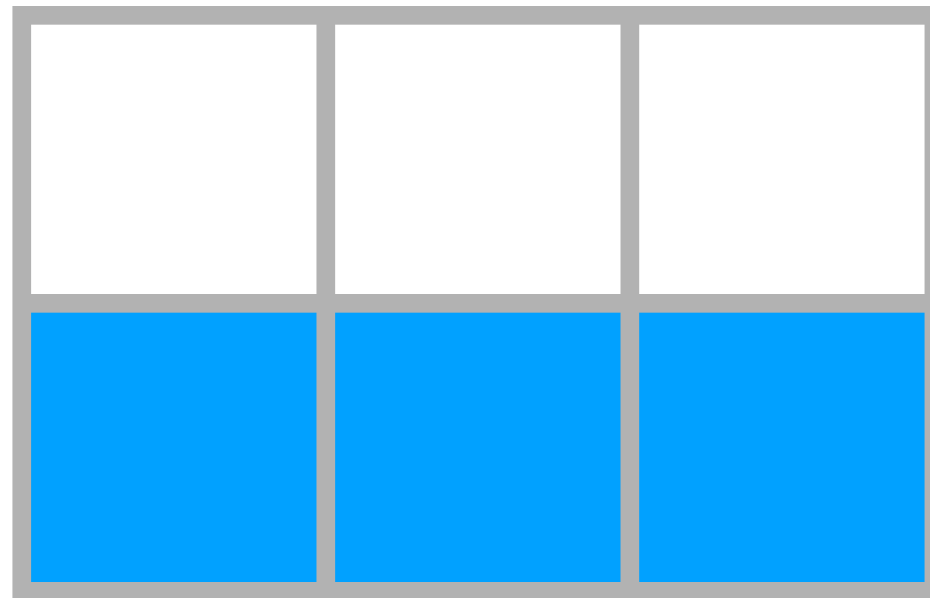


Grid track

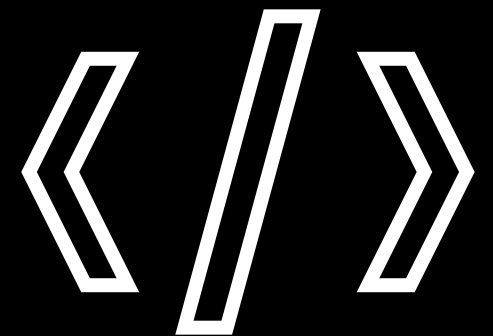
El espacio entre dos **grid lines**. Este espacio puede ser horizontal o vertical.

Puedes pensar en ellos como las columnas o filas del grid.

Grid column: Grid track vertical / **Grid row**: Grid track horizontal



Live coding!



Live coding!

pero antes...

Firefox Quantum DE



Esta última versión de Firefox DE ¡incluye algunas de las mejores herramientas de desarrollo CSS que existen! En particular la herramienta de layout con CSS Grid. Nos va a ayudar a entender y desarrollar nuestro grid de una forma mas ágil.

¡Gracias Mozilla!

Ejemplos realizados

Demo 01 - Creación de grid simple

<https://codepen.io/klaufel/pen/PQoQWz>

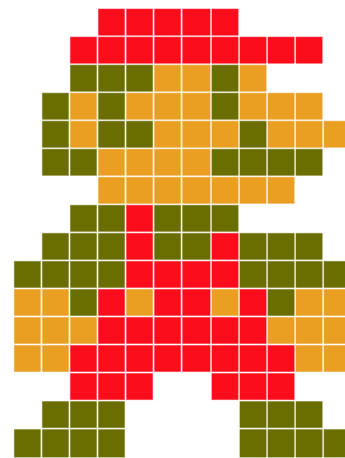
Demo 02 - Grid simple con reordenación de grid tracks

<https://codepen.io/klaufel/pen/yvLvMv>

Demo 03 - Estructura de grid basada en áreas

<https://codepen.io/klaufel/pen/vdYdmO>

¡Extra!

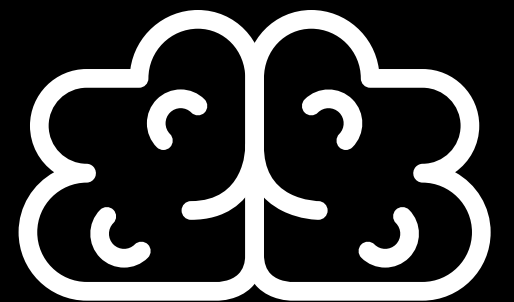


Here we go!! Otra locura más...

**Una prueba de las posibilidades que nos brinda Grid layout.
Desarrollo de una rejilla para crear ¡Pixel art!**

<https://codepen.io/klaufel/pen/opOpXX>

Propiedades



Glosario de propiedades

Propiedades Grid container

```
grid-template-columns  
grid-template-rows  
grid-template-areas  
grid-template  
  
grid-auto-columns  
grid-auto-rows  
grid-auto-flow  
  
grid-column-gap  
grid-row-gap  
grid-gap  
  
grid
```

Propiedades Grid item

```
grid-row-start  
grid-row-end  
grid-row  
  
grid-column-start  
grid-column-end  
grid-column  
  
grid-area
```

Funciones Funciones grid layout

```
repeat()  
  
minmax()
```

**¡A experimentar
se ha dicho!**



Referencias

Guía completa sobre CSS Grid layout (CSS-tricks)

<https://css-tricks.com/snippets/css/complete-guide-grid/>

Curso CSS Grid (4 horas - 25 vídeos)

<https://cssgrid.io/>

Curso Grid práctico por Juan Andrés Núñez en español

<https://www.youtube.com/watch?v=kqB7XD0d0gA>

CSS Grid explicado con ejemplos

<https://gridbyexample.com/examples/>

Listado de pens utilizando CSS Grid (Codepen)

<https://codepen.io/search/pens?q=CSS+Grid&limit=all&type=type-pens>

Introducción a CSS Grid layout (Firefox devtools)

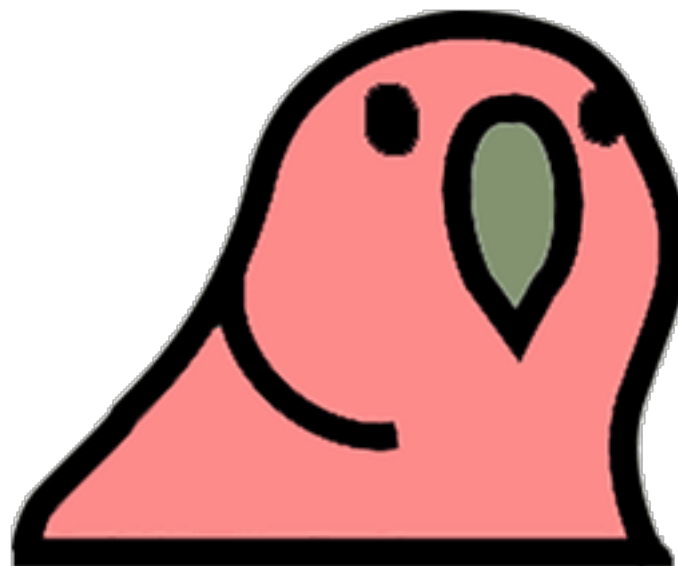
<https://mozilladevelopers.github.io/playground/css-grid/>



¡Muchas gracias!

@klaufel

Frontendly greetings!!!



<spam> obligatorio en todos los mitap! </spam>