

Grids



CSS Grid

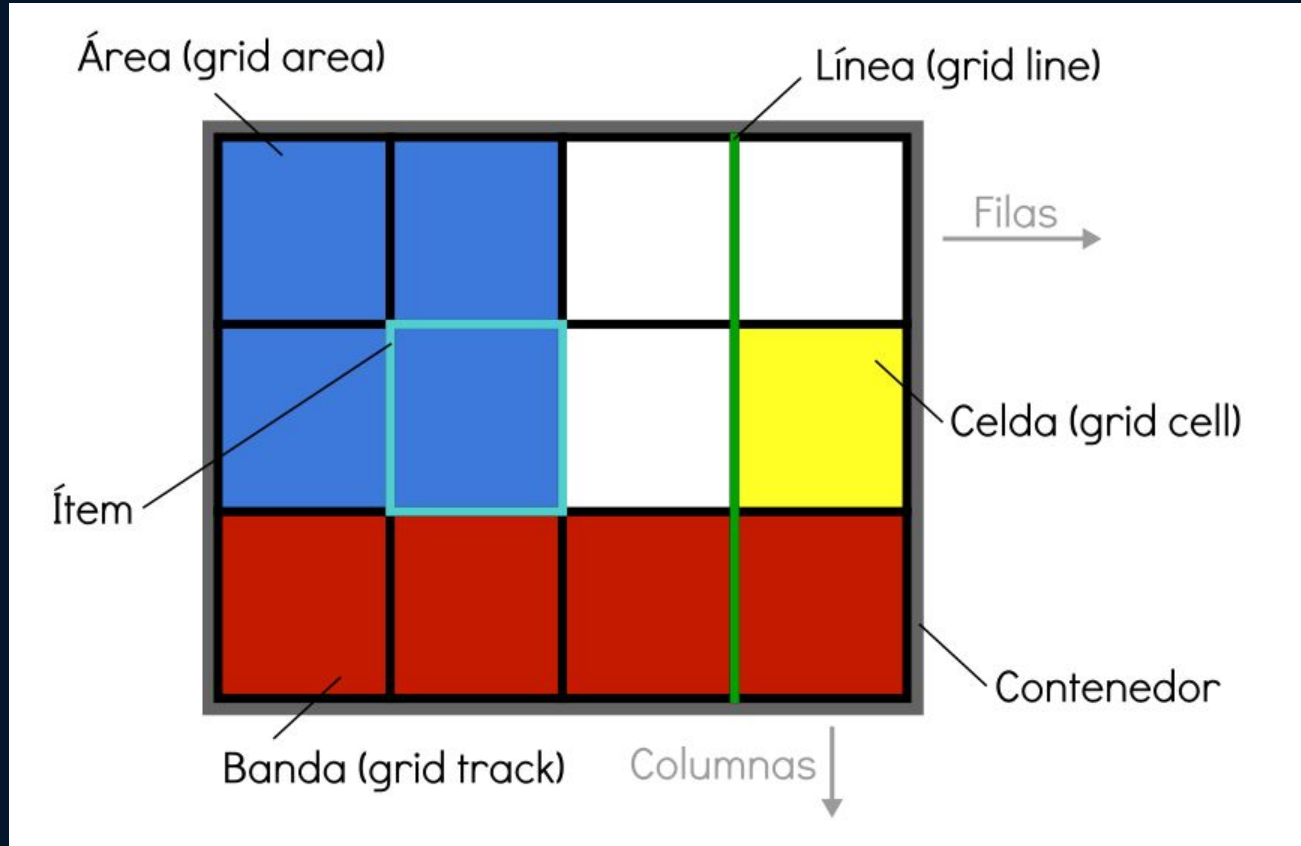
Grids

CSS Grid es un sistema de maquetación web que divide la página en una cuadrícula o rejilla (grid) a partir de la cual se pueden posicionar los diferentes elementos de manera más sencilla, versátil y coherente.

Su practicidad y sus múltiples ventajas lo han convertido en un estándar. Es decir, casi cualquier navegador soporta e interpreta este tipo de código.

La irrupción de CSS Grid, junto con Flexbox, supuso una revolución en el mundo de la programación web, ya que permitía realizar con mucho menos código elementos y estructuras que resultaban muy complejas o directamente imposibles.

Grids



Grids

Contenedor: El elemento padre contenedor que definirá la cuadrícula o rejilla.

Ítem: Cada uno de los hijos que contiene la cuadrícula (elemento contenedor).

Celda (grid cell): Cada uno de los cuadritos (unidad mínima) de la cuadrícula.

Area (grid area): Región o conjunto de celdas de la cuadrícula.

Banda (grid track): Banda horizontal o vertical de celdas de la cuadrícula.

Línea (grid line): Separador horizontal o vertical de las celdas de la cuadrícula.

Grids

Para utilizar cuadrículas Grid CSS, trabajaremos bajo el siguiente escenario:

```
<div class="grid"> <!-- contenedor -->
  <div class="a">Item 1</div> <!-- cada uno de los ítems del grid -->
  <div class="b">Item 2</div>
  <div class="c">Item 3</div>
  <div class="d">Item 4</div>
</div>
```

Grids

Ventajas de CSS Grid:

Mucha más flexibilidad: te permite controlar los elementos en las 2 dimensiones y con total libertad.

Menos código y menos bugs: se reduce considerablemente el código empleado, lo que conlleva revisiones más sencillas y menor probabilidad de bugs.

Optimización de recursos: al ser código más simple y consistente se economiza tanto en tiempo como en recursos necesarios para mostrar la página.

Responsive más sencillo: permite crear elementos dinámicos que se adaptan a diferentes tamaños o resoluciones sin complicaciones.

Grids

Como aplicar CSS Grid:

Para activar la cuadrícula grid hay que utilizar sobre el elemento contenedor la propiedad display y especificar el valor grid o inline-grid.

Tipo de elemento	Descripción
inline-grid	Establece una cuadrícula con ítems en línea, de forma equivalente a inline-block.
grid	Establece una cuadrícula con ítems en bloque, de forma equivalente a block.

Grids

Como aplicar CSS Grid:

Para activar la cuadrícula grid hay que utilizar sobre el elemento contenedor la propiedad display y especificar el valor grid o inline-grid.



Este valor influye en como se comportará la cuadrícula con el contenido exterior. El primero de ellos permite que la cuadrícula aparezca encima/debajo del contenido exterior (en bloque) y el segundo de ellos permite que la cuadrícula aparezca a la izquierda/derecha (en línea) del contenido exterior (ojo, la cuadrícula entera, no cada uno de sus ítems)

Tipo de elemento	Descripción
inline-grid	Establece una cuadrícula con ítems en línea, de forma equivalente a inline-block.
grid	Establece una cuadrícula con ítems en bloque, de forma equivalente a block.

Grids

Grid con filas y columnas:

Es posible crear cuadrículas con un tamaño explícito. Para ello, sólo tenemos que usar las propiedades CSS `grid-template-columns` y `grid-template-rows`, que sirven para indicar las dimensiones de cada celda de la cuadrícula, diferenciando entre columnas y filas. Las propiedades son las siguientes:

Propiedad	Valor	Descripción
<code>grid-template-columns</code>	<code>[<u>col1</u>] [<u>col2</u>] ...</code>	Establece el  de cada columna (<u>col 1</u> , <u>col 2</u> ...).
<code>grid-template-rows</code>	<code>[<u>fila1</u>] [<u>fila2</u>] ...</code>	Establece el  de cada fila (<u>fila 1</u> , <u>fila 2</u> ...).

Grids

Grid con filas y columnas:

```
.grid {  
  display: grid;  
  grid-template-columns: 50px 300px 1fr;  
  grid-template-rows: 200px 2fr 5% 1fr;  
}
```

Con estas propiedades definimos el numero de columnas(grid-template-columns) y filas(grid-template-rows). Cada valor que indiquemos será una columna o una fila. En el ejemplo tenemos 3 columnas y 4 filas. La unidad fraccionaria (fr) es una unidad relativa que nos permite definir un cierto porcentaje de la totalidad del contenedor dependiendo de los valores que ingresemos ejemplo, si tengo dos columnas (1fr, 3fr) se debe sumar las dimensiones que será la totalidad del contenedor y la primera columna ocupara solo la cuarta parte y la segunda columna ocupara tres cuartos de la totalidad del contenedor.

Espacios en Grids

Row-Gap:

Con esta propiedad podemos definir espacios entre filas solo debemos indicar un valor numerico, ejemplo `rows-gap:10px;`

Columns-Gap:

Con esta propiedad podemos definir espacios entre columnas solo debemos indicar un valor numerico, ejemplo `columns-gap:10px;`

Gap: Con esta propiedad podemos definir espacios entre filas y columnas solo debemos indicar un valor numero ya sea en unidades relativas (% ,em, rem,etc) y absolutas(px,etc). Esta propiedad nos permite definir un valor para los espacios de filas y otro para columnas.
ejemplo `gap:10px 20px;`

Grid-template-areas

Grid-template-areas:

Define una plantilla de cuadrícula haciendo referencia a los nombres de las áreas de cuadrícula que se especifican con la propiedad grid-area. La repetición del nombre de un área de la cuadrícula hace que el contenido abarque esas celdas. Un punto significa una celda vacía. La sintaxis en sí misma proporciona una visualización de la estructura de la cuadrícula.