**1- De dónde venimos**

* Una única columna / cuerpo de contenido
* Tablas.
  + Con imágenes GIF (spacers).
* CSS Moderno
  + Estilos externos.
* Floats.
  + Hack.
* RWD.
  + Diversidad de dispositivos.
* Flexbox.
* **CSS GRID.**

### 2- Qué es CSS GRID

CSS GRID es un **nuevo módulo de posicionamiento CSS**, al igual que FlexBox y BlockBox.

CSS GRID permite crear rejillas descritas en CSS y que se comportan de forma independiente de su emplazamiento en el HTML. Así,**CSS GRID permite separar el orden de definición en el código fuente de su representación visual**.

CSS GRID puede convertirse en el nuevo estándar para la maquetación Web (junto con FlexBox) y es [aceptado por la mayoría de navegadores Web](http://caniuse.com/#search=GRID).

### 3- CSS GRID vs FlexBox

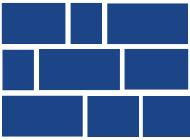
El módulo de posicionamiento CSS GRID no sustituye a FlexBox sino que**lo complementa**, como iremos viendo.

FlexBox



* 1 dimensión.
* Document flow normal.

CSS GRID



* 2 dimensiones.
* Líneas y áreas GRID.
* Alineado en filas y columnas.

### 4- Conceptos esenciales

Junto con la nueva especificación del módulo CSS GRID vienen algunos términos y conceptos que hay que entender para poder comenzar a usarlo.

**GRID Container & GRID Item**

El elemento que hace de rejilla/contenedor y cada uno de los items que se muestran dentro. Esta relación parent/children es llamada ***GRID context***.

/\* Declarando un GRID container \*/

main {

display: grid;

}

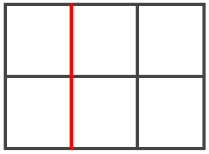
### 4- Conceptos esenciales

**GRID Line**

El contenedor GRID está formado por dos tipos de líneas: horizontales y verticales.

Las líneas horizontales corren a través del eje X del contenedor GRID y las verticales corren a través del eje Y.

Podemos hacer referencia a una GRID line por número o por nombre.



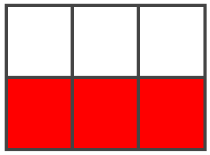
### 4- Conceptos esenciales

**GRID Track**

Nombre genérico para un GRID row o GRID column.

Es el espacio vertical u horizontal entre dos GRID lines **consecutivas**.

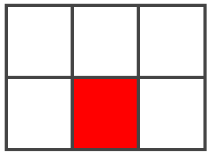
Define la altura de un GRID column o la anchura de una GRID row.



**GRID Cell**

La **intersección** de un GRID row y un GRID column es llamada GRID cell, es decir, el espacio **entre** 4 GRID lines.

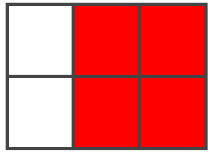
Es, además, la unidad más pequeña a nuestra disposición para colocar un GRID item.



**GRID Area**

Es **cualquier porción** del GRID contenida entre 4 GRID lines.

Puede contener N número de GRID cells.



### Demo 1

Vamos a realizar una primera práctica con CSS GRID para utilizar algunas de sus características esenciales:

Código [disponible en GitHub](https://github.com/juanwmedia/css-grid-lo-que-debes-saber).

* La unidad fr.
* Declarar y definir un GRID.
* Especificar GRID tracks como filas y columnas.
* Alinear el GRID container.
* Implicit GRID.

**Declarar y definir un GRID**

Define el contenedor GRID. [*inline-grid*](http://codepen.io/rachelandrew/pen/BNXyQa)también disponible.

/\* Declarando un GRID container \*/

main {

display: grid;

}

**Especificar GRID tracks como filas y columnas**

*grid-template-rows & columns*define las filas y columnas del GRID container.

main {

display: grid;

grid-template-rows: 100px 100px 100px;

grid-template-columns: 33% 33% 100px;

}

Código [disponible en GitHub](https://github.com/juanwmedia/css-grid-lo-que-debes-saber)

### Demo 1

**Unidad fr**

Representa **una fracción del espacio disponible en GRID container**. Utilizar esta medida permite que GRID responda y se adapte a los cambios en el viewport.

main {

...

grid-template-columns: 2fr 1fr 1fr;

}

Se puede utilizar en combinación con otras unidades para crear GRID tracks flexibles junto con GRID tracks fijos.

main {

...

grid-template-columns: 2fr 1fr 300px;

}