



PERFECCIÓN  
**DIGITAL**



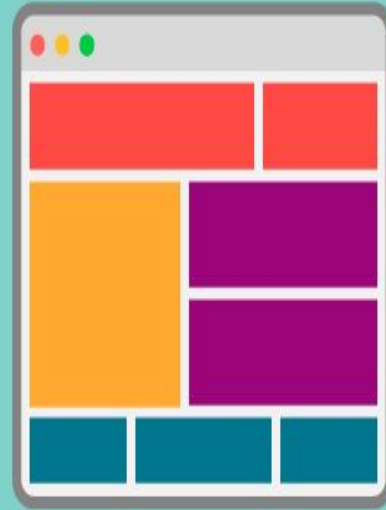
# Introducción a GridLayout

**Ladislao Ch.**  
Expositor



# Agenda

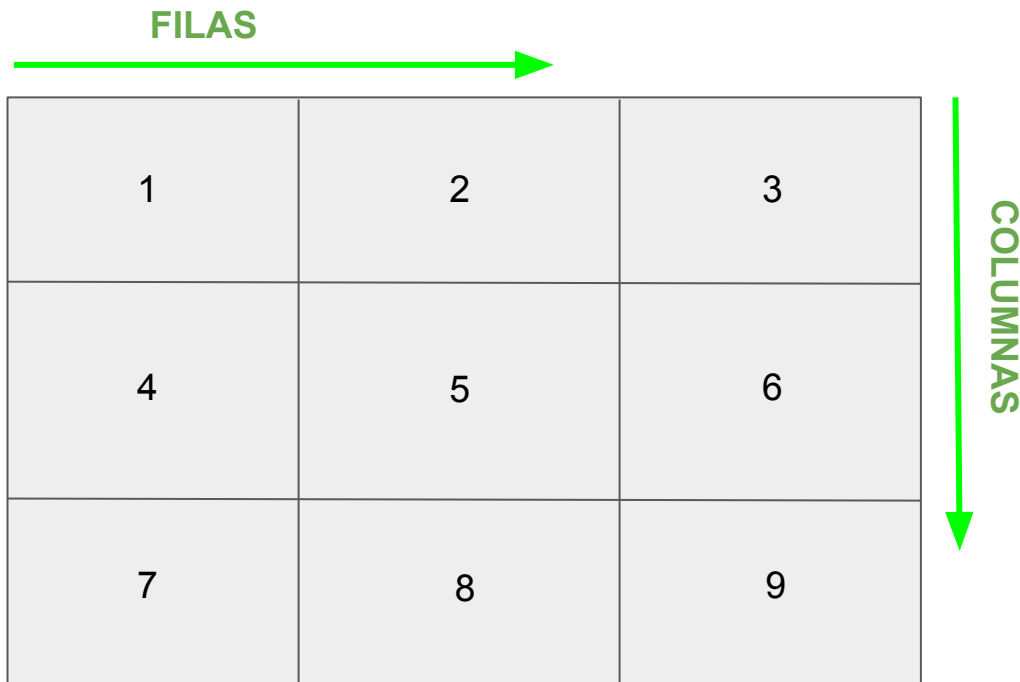
- ¿Qué es Gridlayout?
- Ventajas
- Conceptos generales
- Propiedades



# ¿Qué es Gridlayout?

Es un modelo de layout que permite la construcción de grillas (o también llamadas cuadrículas) de forma dinámica.

Es Bidimensional (2 dimensiones) porque se trabaja con Filas y Columnas.



Gridlayout trabaja también con elementos Padres e hijos.

### **Grid Container : (Padre)**

Es una etiqueta que hace la función de contenedor o **elemento Padre** que contiene o encierra dentro de él varios elementos (**grid items**).

Este elemento Padre se caracteriza por tener asignado **display : grid;**

### **Grid item : (Hijo)**

Son los elementos que componen la cuadrícula o grilla en el contenedor Padre.

```
<div class="grid-container">  
  <div class="grid-item">A</div>  
  <div class="grid-item">B</div>  
  <div class="grid-item">C</div>  
  <div class="grid-item">D</div>  
  <div class="grid-item">E</div>  
</div>
```

# Ventajas de Gridlayout

- Facilita diseños en 2D al manejar Filas y columnas.
- Es perfecto para construir diseños asimétricos o poco ortodoxos.
- Cada celda es dinámica. Es decir, le podemos dar el tamaño que queramos y el resto de celdas se adaptará automáticamente.
- Podemos indicar donde va a empezar y terminar cada celda en su respectiva fila o columna.

# Conceptos Generales

## **Grid Line:**

Son líneas imaginarias o invisibles divisoras horizontales y verticales dentro de la cuadrícula.  
Estas son las “**column line**” o “**row line**”

## **Grid Track:**

Es un espacio entre dos líneas adyacentes, comúnmente llamadas **filas** y **columnas**.

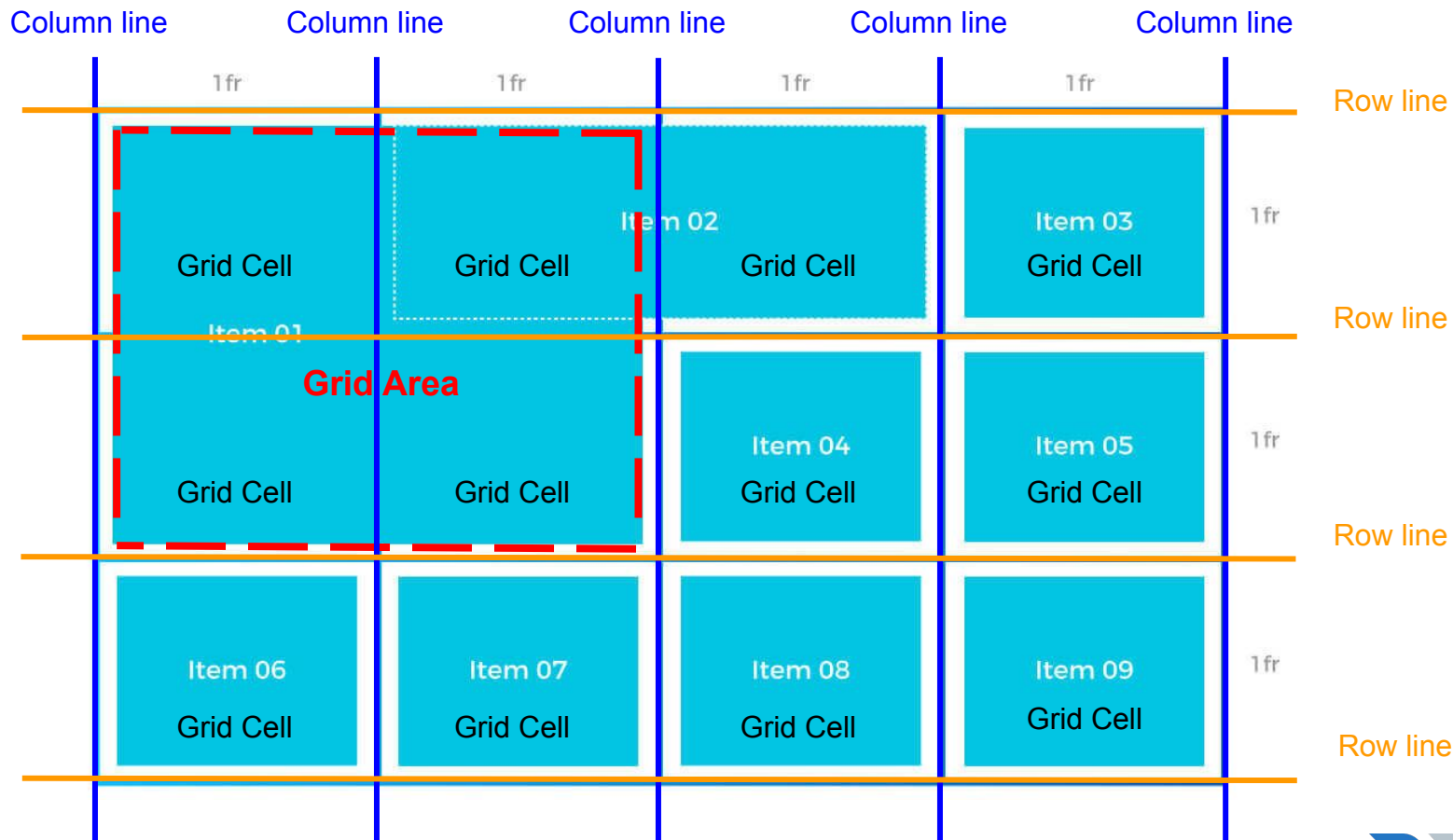
## **Grid Cell:**

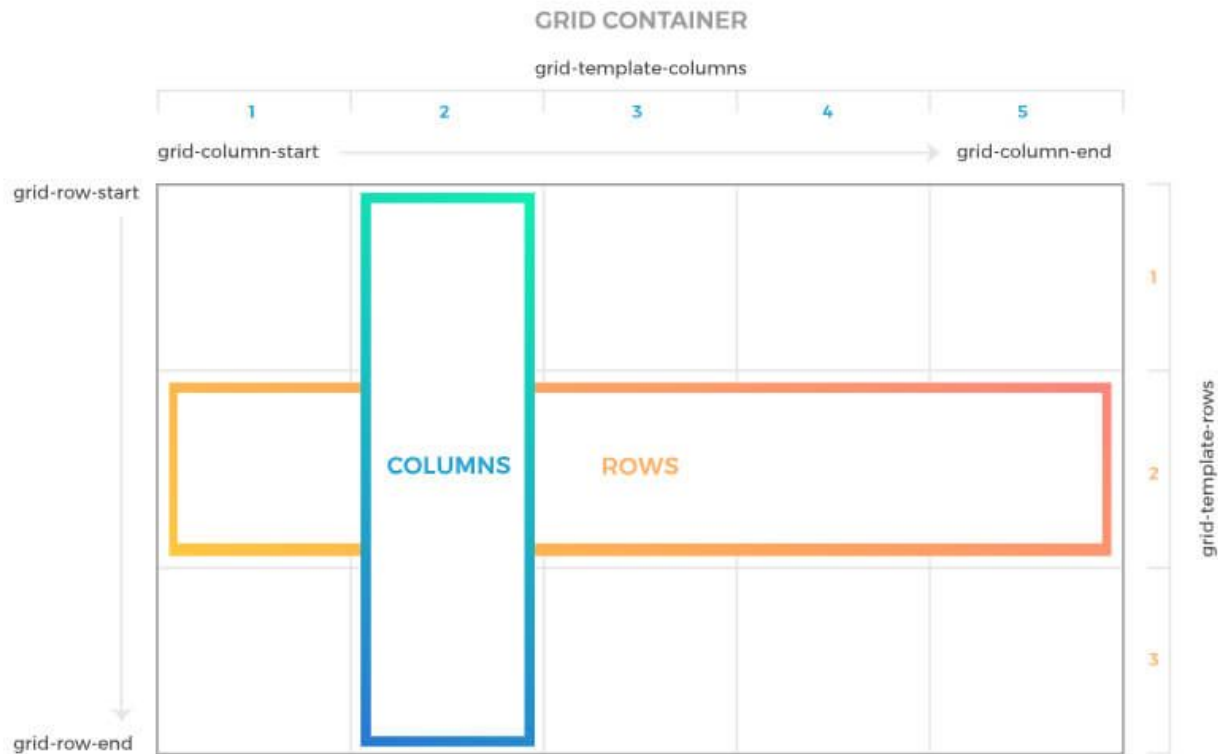
Es cada celda de la cuadrícula o grilla.

## **Grid Area:**

Es un espacio rodeado por 4 *Grid Lines*, el cual, puede ser desde 1 celda a varias.







# Propiedades

## Grid Container (elemento Padre)

1. display : grid
2. grid-template-columns
3. grid-template-rows
4. grid-template
5. Medidas :  
fr y auto
6. column-gap  
(antes : grid-column-gap)
7. row-gap  
(antes : grid-row-gap)
8. gap  
(antes : grid-gap)
9. Funcion :  
repeat()

# Propiedades

## **Grid Item (elemento Hijo)**

1. grid-column-start
2. grid-column-end
3. grid-column
4. grid-row-start
5. grid-row-end
6. grid-row

# Propiedades - Padre

## **display : grid**

- Vuelve al elemento Padre una Grilla o cuadrícula y a sus elementos hijos celdas de esa grilla.
- Por defecto te da una sola columna sino se le indica la cantidad.

# Propiedades - Padre

## **grid-template-columns**

- Indicas la cantidad de columnas en la grilla o cuadrícula y la medida para cada columna.

## **grid-template-rows**

- Indicas la cantidad de filas en la grilla o cuadrícula y la medida para cada fila.

## **grid-template**

- Es un **shortcut** (forma corta), Indica la cantidad de filas y columnas en una sola propiedad y su medida para cada fila y columna.

# Propiedades - Padre

## Medidas :

### **FR** (fracción)

- Son las ' n ' fracciones del espacio disponibles después de establecer las medidas fijas. (medidas en proporción).

### **Auto**

- Es el espacio faltante ,si hubiera, y siempre el espacio de su contenido.

# Propiedades - Padre

## column-gap

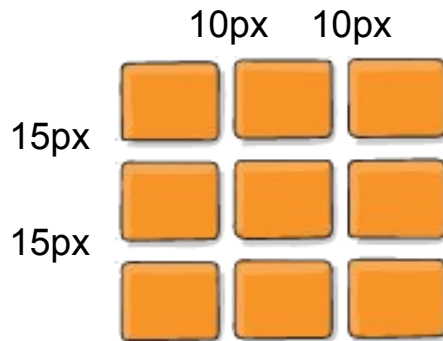
- Da espacio solo entre las columnas.

## row-gap

- Da espacio solo entre las filas.

## gap

- Es un **shortcut** (forma corta), Da el mismo espacio entre filas y columnas.





# Propiedades - Padre

## **Función : Repeat()**

- Función usada para la repetición de medidas en las filas o columnas.
- Recibe 2 parámetros :

1° cantidad de veces a repetirse(fila o columna)

2° medidas para las filas o columnas a repetirse

# Propiedades - Hijo

## **grid-column-start**

- Indica en qué column-grid empieza el elemento.

## **grid-column-end**

- Indica en qué column-grid termina el elemento.

## **grid-column**

- Es un **shortcut** (forma corta), Indica el inicio y fin en los column-grid.

## **grid-row-start**

- Indica en qué row-grid debe empezar el elemento.

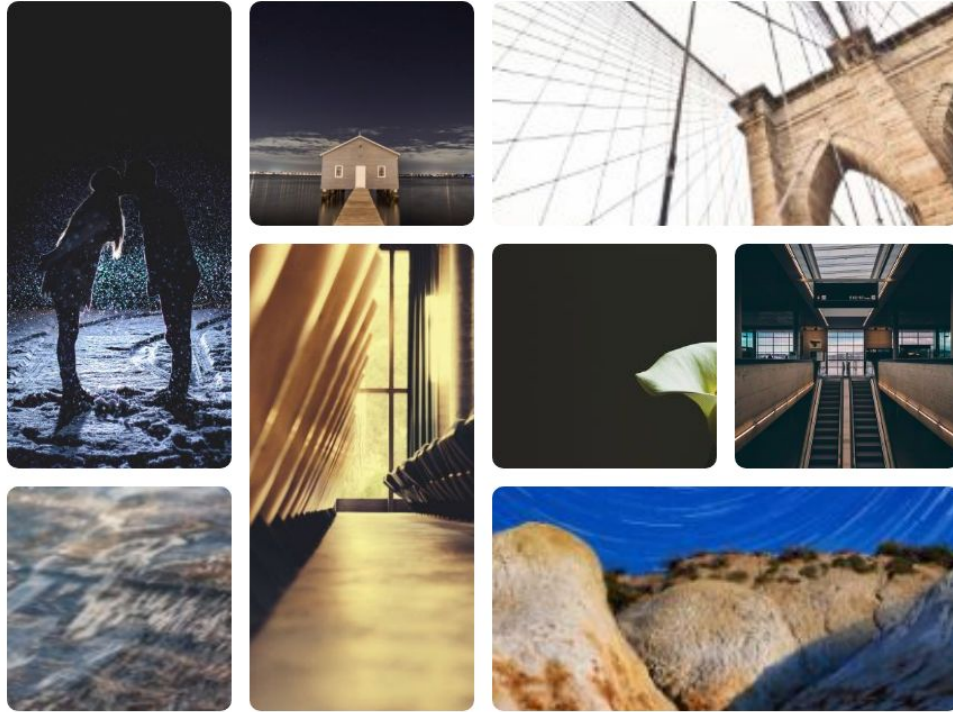
## **grid-row-end**

- Indica en qué row-grid debe terminar el elemento.

## **grid-row**

- Es un **shortcut** (forma corta), Indica el inicio y fin en los row-grid.

# Galeria - Ejemplo



# ¡GRACIAS!



IG: @perfecci0ndigital

FB: @perfecci0ndigital