

**PROGRAMA DE ESTUDIOS**

# **DESARROLLO DE SISTEMAS DE INFORMACIÓN**

**FRONTEND ORIENTADO  
A SERVICIOS**

**Tema**

**SISTEMA DE CUADRICULA CON CSS  
GRID PARTE I**

## SISTEMA DE CUADRICULA CON CSS GRID PARTE I

## 1. Introducción a CSS Grid

CSS Grid es un sistema de diseño en CSS que permite crear layouts complejos y flexibles utilizando una cuadrícula. Es especialmente útil para organizar los elementos de una página en filas y columnas, facilitando la creación de diseños estructurados y adaptables.

A diferencia de Flexbox, que es ideal para diseños unidimensionales (en una sola fila o columna), CSS Grid es bidimensional, permitiendo trabajar con filas y columnas de forma simultánea.

## 2. Definir Filas y Columnas

Para utilizar CSS Grid, primero debemos declarar un contenedor como `display: grid`. Luego, usamos propiedades como `grid-template-columns` y `grid-template-rows` para definir el número y tamaño de las columnas y filas dentro de la cuadrícula.

- `grid-template-columns`: Define el número y el tamaño de las columnas.
- `grid-template-rows`: Define el número y el tamaño de las filas.

**Ejemplo Básico:**

```
.container {  
  display: grid;  
  /* Tres columnas iguales */  
  grid-template-columns: 1fr 1fr 1fr;  
  /* Dos filas de diferentes tamaños */  
  grid-template-rows: 100px 200px;  
}
```

En este ejemplo:

- 1fr representa una fracción del espacio disponible, repartiendo el ancho entre tres columnas iguales.
- Las filas tienen alturas específicas: la primera de 100px y la segunda de 200px.

### 3. Colocación de Elementos en la Cuadrícula

CSS Grid nos permite posicionar elementos en lugares específicos de la cuadrícula. Utilizamos propiedades como `grid-column` y `grid-row` para controlar en qué lugar se posicionará cada elemento.

- `grid-column`: Permite especificar en qué columna comienza y termina el elemento.
- `grid-row`: Permite especificar en qué fila comienza y termina el elemento.

**Ejemplo de Colocación:**

```
.item1 {  
  /* Ocupa desde la columna 1 hasta la columna 3 */  
  grid-column: 1 / 3;  
  /* Ocupa solo la primera fila */  
  grid-row: 1 / 2;  
}
```

### Ejemplo Completo: Crear una Cuadrícula Básica

Vamos a crear una cuadrícula con 3 columnas y 2 filas, y posicionaremos algunos elementos dentro de esta cuadrícula.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Ejemplo de CSS Grid</title>
  <style>
    .container {
      display: grid;
      grid-template-columns: 1fr 2fr 1fr; /* Tres columnas de diferente tamaño */
      grid-template-rows: 150px 150px; /* Dos filas de igual tamaño */
      gap: 10px; /* Espacio entre elementos */
    }
    .item1 { background-color: lightcoral; grid-column: 1 / 3; }
    .item2 { background-color: lightblue; grid-column: 2 / 4; grid-row: 1 / 3; }
    .item3 { background-color: lightgreen; grid-column: 1 / 2; grid-row: 2 / 3; }
    .item4 { background-color: lightgoldenrodyellow; }
  </style>
</head>
<body>
  <div class="container">
    <div class="item1">Elemento 1</div>
    <div class="item2">Elemento 2</div>
    <div class="item3">Elemento 3</div>
    <div class="item4">Elemento 4</div>
  </div>
</body>
</html>
```

### Explicación del Ejemplo:

- La cuadrícula tiene tres columnas y dos filas.
- Usamos grid-column y grid-row para posicionar elementos:
  - item1 ocupa las primeras dos columnas de la primera fila.
  - item2 ocupa desde la segunda columna hasta la tercera y ambas filas.
  - item3 está en la primera columna de la segunda fila.
  - item4 ocupa el lugar restante.

## Ejercicios Prácticos

### Ejercicio 1: Crear una Cuadrícula de Tarjetas

#### Instrucciones:

1. Crea una cuadrícula con 3 columnas y 3 filas.
2. Coloca nueve elementos en la cuadrícula y dale a cada uno un color de fondo distinto.
3. Asegúrate de que cada elemento ocupa una sola celda.

### Ejercicio 2: Disposición de un Blog con CSS Grid

#### Instrucciones:

1. Crea una cuadrícula que simule un diseño de blog con tres secciones:
  - Un encabezado que ocupe toda la primera fila.
  - Una sección de contenido principal que ocupe dos columnas en la segunda fila.
  - Una barra lateral que ocupe una columna al lado derecho del contenido principal en la segunda fila.
  - Un pie de página en la tercera fila que ocupe todo el ancho.
2. Utiliza grid-template-areas para organizar las secciones.



INSTITUTO  
**KHIPU**