**Paso 1: Qué é CSS Grid?**

CSS Grid é un sistema de disposición (layout) en CSS que permite organizar o contido dunha páxina en filas e columnas, creando unha cuadrícula. A diferencia do modelo tradicional de **flexbox**, CSS Grid permite controlar o posicionamento de elementos en dúas dimensións: horizontal e vertical.

**Paso 2: Crear a estrutura básica**

Imos empezar por configurar a nosa páxina HTML e aplicar un estilo básico con CSS. Primeiro, crea un ficheiro HTML básico onde definiremos a estrutura de columnas e filas. Aquí, imos usalo para crear as "caixas" que simulan as seccións do cadro.

**HTML (estruturas a cuadrícula)**

html

Copiar código

<!DOCTYPE html>

<html lang="gl">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Replicando Mondrian</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="grid-container">

<div class="caixa caixa-1"></div>

<div class="caixa caixa-2"></div>

<div class="caixa caixa-3"></div>

<div class="caixa caixa-4"></div>

<div class="caixa caixa-5"></div>

<div class="caixa caixa-6"></div>

<div class="caixa caixa-7"></div>

</div>

</body>

</html>

Nese código, creamos un **div** principal chamado grid-container e dentro disto, varios **div** secundarios que serán as "caixas" que compoñen o cadro. Cada caixa terá clases diferentes para poder aplicar estilos diferentes.

**CSS (estilizando a cuadrícula)**

Agora imos crear un arquivo CSS que nos axude a establecer o sistema de grid e tamén a cor das caixas. O importante neste paso é aprender como establecer as **filas e columnas** e como asignarlles dimensións.

css

Copiar código

/\* Establecemos un tamaño de borde básico para toda a páxina \*/

body {

margin: 0;

padding: 0;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

background-color: #f0f0f0;

}

/\* O contedor que terá a nosa cuadrícula \*/

.grid-container {

display: grid; /\* Activa a cuadrícula \*/

grid-template-columns: repeat(3, 100px); /\* Definimos 3 columnas de 100px cada unha \*/

grid-template-rows: repeat(3, 100px); /\* Definimos 3 filas de 100px cada unha \*/

gap: 10px; /\* Definimos un espazo de 10px entre cada caixa \*/

width: 350px; /\* Largura total da cuadrícula \*/

height: 350px; /\* Altura total da cuadrícula \*/

background-color: #ffffff; /\* Cor de fondo da cuadrícula \*/

border: 2px solid black; /\* Borde ao redor da cuadrícula \*/

}

/\* Estilos básicos para as caixas \*/

.caixa {

width: 100%;

height: 100%;

background-color: #cccccc;

border: 1px solid #333;

}

/\* Estilos para cada caixa para dar cores diferentes \*/

.caixa-1 { background-color: red; }

.caixa-2 { background-color: blue; }

.caixa-3 { background-color: yellow; }

.caixa-4 { background-color: white; }

.caixa-5 { background-color: black; }

.caixa-6 { background-color: blue; }

.caixa-7 { background-color: yellow; }

Neste CSS, fixamos unha cuadrícula de 3 filas e 3 columnas, cada unha cunha anchura e altura de 100px, e un espazo de 10px entre elas. Usamos o grid-template-columns e grid-template-rows para definir a estrutura de filas e columnas. Ademais, a gap serve para separar as caixas entre si.

**Paso 3: Modificar a cuadrícula para que se pareza a un cadro de Mondrian**

Agora que temos a cuadrícula, podemos asignar as caixas ás posicións correctas para que se parezan ao estilo de Mondrian. A disposición das caixas na cuadrícula pode facerse mediante o sistema de **posicións do grid**. Usamos **grid-column** e **grid-row** para decidir onde colocar cada caixa dentro da cuadrícula.

Vamos modificar un pouco o CSS para asignar as caixas a posicións específicas:

css

Copiar código

/\* A cuadrícula 3x3 mantense igual, pero agora asignaremos posicións específicas a cada caixa \*/

.grid-container {

display: grid;

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

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

gap: 10px;

}

/\* Modificamos as caixas para que se poidan posicionar correctamente \*/

.caixa-1 {

background-color: red;

grid-column: 1 / 3; /\* Empeza na columna 1 e vai ata a columna 3 \*/

grid-row: 1 / 2; /\* Empeza na fila 1 e vai ata a fila 2 \*/

}

.caixa-2 {

background-color: blue;

grid-column: 3 / 4; /\* Colócase na columna 3 \*/

grid-row: 1 / 2;

}

.caixa-3 {

background-color: yellow;

grid-column: 1 / 2;

grid-row: 2 / 3;

}

.caixa-4 {

background-color: white;

grid-column: 2 / 3;

grid-row: 2 / 3;

}

.caixa-5 {

background-color: black;

grid-column: 3 / 4;

grid-row: 2 / 3;

}

.caixa-6 {

background-color: blue;

grid-column: 1 / 2;

grid-row: 3 / 4;

}

.caixa-7 {

background-color: yellow;

grid-column: 2 / 4;

grid-row: 3 / 4;

}

**Paso 4: Explicación das propiedades importantes**

* **grid-template-columns** e **grid-template-rows**: Usamos estas propiedades para definir como será a cuadrícula. Aquí definimos 3 columnas e 3 filas, cada unha de 100px.
* **grid-column** e **grid-row**: Estas propiedades as usamos para dicir onde queremos que se colóquen as caixas na cuadrícula. Pódense definir os comezos e os finais das columnas ou filas.
* **gap**: O espazo entre as caixas. Aquí fixámolo de 10px.

**Paso 5: Proba e mellora**

Unha vez que vexas a cuadrícula, verás que o patrón comeza a parecerse a algo do estilo de Mondrian, con rectángulos de diferentes cores. Agora podes mellorar o cadro aumentando o número de filas e columnas, ou axustando as cores ou as dimensións das caixas para lograr unha mellor semellanza co cadro orixinal.

Este é un exercicio perfecto para practicar e mellorar no uso de **CSS Grid**. Experimenta movendo as caixas e cambiando as cores para afinar o teu traballo!

**Resumo de conceptos clave:**

1. **CSS Grid** permite organizar os elementos nunha cuadrícula de filas e columnas.
2. Usamos **grid-template-columns** e **grid-template-rows** para crear a estrutura da cuadrícula.
3. Podemos usar **grid-column** e **grid-row** para asignar posicións exactas aos elementos.
4. As **caixas** ou **elementos** de grid poden ter diferentes cores ou tamaños.