

CSS grid

Objetivos

Aprender que es CSS grid y como funciona

Conocer las propiedades de CSS grid

Fraccionar cuadrículas

Crear un layout

¿Que es CSS grid?

CSS Grid es una tecnología de diseño para sitios web que permite a los desarrolladores crear diseños de páginas web complejos y flexibles utilizando una cuadrícula bidimensional. Con CSS Grid, puedes dividir el espacio de una página web en filas y columnas, y luego colocar los elementos HTML dentro de esta cuadrícula de una manera precisa y controlada.

Algunas características de CSS Grid son:

Cuadrícula bidimensional:

CSS Grid opera en ambas direcciones, lo que facilita la creación de diseños complejos y multidimensionales.

Filas y columnas definidas:

Puedes definir filas y columnas explícitamente, lo que te da un control preciso sobre cómo se distribuyen los elementos en la página.

Diseño responsive:

CSS Grid es especialmente poderoso para crear diseños que se adaptan de manera elegante a diferentes tamaños de pantalla y dispositivos, lo que facilita la creación de diseños responsivos.

Para comprender como funciona CSS grid es necesario conocer el concepto de contenedor y elemento hijo, tal como lo vimos en flexbox grid tiene un grid-container y al interior de este sus grid items.

```
<div
  class="grid-container">
  <div class="grid-item">1</div>
  <div class="grid-item">2</div>
  <div class="grid-item">3</div>
</div>
```

Para crear un grid-container debemos agregar la propiedad display: grid a un elemento padre, de esta manera todos los elementos hijos de este elemento serán grid-items.

```
.grid-container {
  display: grid;
}
```

Al agregar la propiedad display: grid a un elemento padre, este automáticamente se convierte en un grid-container, y todos sus hijos en grid-items. Los grid-items se pueden posicionar en el grid-container usando las propiedades grid-column y grid-row.

Propiedades de CSS grid

CSS grid tiene muchas propiedades que nos permiten crear diseños complejos, a continuación veremos las más importantes.

grid template columns:

La propiedad grid-template-columns es la base de la creación de una cuadrícula en CSS. Al usar esta propiedad, manipulamos dos características muy importantes: el número de columnas que tendrá la cuadrícula y el ancho de las mismas.

Ejemplo:

```
grid-template-columns: repeat(3, 1fr);
```

En el ejemplo anterior dividimos en 3 columnas de un fr (fracción) cada una, si no sabes lo que es un fr no te preocupes mas adelante lo abordaremos.

grid-template-rows:

Es una propiedad de CSS Grid que se utiliza para definir el tamaño y la estructura de las filas dentro de una cuadrícula CSS. Esta propiedad te permite especificar cómo se distribuirán las filas dentro de la cuadrícula y cómo se ajustarán al contenido o al tamaño del contenedor.

Ejemplo:

```
grid-template-rows: valor1 valor2 valor3 ...;
```

Cada "valor" representa una fila dentro de la cuadrícula y puede ser una longitud (por ejemplo, píxeles, porcentajes, unidades fr o contenido), una palabra clave (como "auto" o "min-content"), o una fracción del espacio disponible (usando la unidad fr).

"grid-template-rows" se combina con otras propiedades de CSS Grid, como "grid-template-columns", para establecer la estructura completa de la cuadrícula y posicionar los elementos dentro de ella de manera flexible y precisa.

grid-template-areas:

"grid-template-areas" es una propiedad de CSS Grid que proporciona una forma visual y declarativa de definir la estructura de una cuadrícula CSS y cómo los elementos HTML se distribuirán dentro de ella. Con "grid-template-areas", puedes asignar nombres a áreas específicas dentro de la cuadrícula y luego colocar los elementos HTML en esas áreas utilizando esos nombres.

El valor de grid-template-areas es una representación visual de la cuadrícula utilizando nombres de áreas y caracteres especiales como comillas ("). Cada línea en la declaración de grid-template-areas representa una fila de la cuadrícula, y los nombres de las áreas separados por espacios indican las celdas en esa fila. Los puntos (.) representan celdas vacías es decir, no tienen ningún elemento asignado a ellas. Sirven para indicar que no se desea colocar ningún elemento en esa posición específica.

Ejemplo:

```
grid-template-areas:  
  "header header header"  
  "main main sidebar"  
  "footer footer footer";
```

En el ejemplo anterior creamos una cuadrícula de 3 filas y 3 columnas, en la primera fila tenemos 3 columnas con el nombre de header, en la segunda fila tenemos 2 columnas con el nombre de main y sidebar, y en la tercera fila tenemos 3 columnas con el nombre de footer.

grid-template:

La propiedad grid-template es una propiedad abreviada de CSS Grid que se utiliza para establecer todas las propiedades de plantilla de una cuadrícula CSS en una sola declaración. Esto incluye grid-template-rows, grid-template-columns y grid-template-areas.

Ejemplo:

```
grid-template:
  grid-template-rows/grid-template-columns | grid-template-areas;
```

Otro ejemplo:

```
.grid-container {
  display: grid;
  grid-template:
    Definir el área y tamaño de la cabecera
    "header header header" 100px
    Definir el área y tamaño de la barra lateral y el contenido
    "sidebar content content" 1fr
    Definir el área y tamaño del pie de página
    "footer footer footer" 50px;
}
```

¿Que es un fr?

La unidad relativa fr en CSS o unidad de medida fr la introduce el sistema de posicionamiento CSS Grid. Esta unidad representa una fracción del contenedor. es decir representa una fracción del espacio disponible en el contenedor de la cuadrícula. Cuando se utiliza 1fr, significa que esa columna o fila ocupará una fracción igual del espacio disponible. Por ejemplo, si tienes una cuadrícula con tres columnas y asignas 1fr a cada una de ellas, cada columna ocupará un tercio del ancho total disponible.

Definiendo espaciado

Una propiedad de CSS Grid que se utiliza para establecer el espacio entre filas y columnas en una cuadrícula es el gap. También es conocido como "espaciado de rejilla". El gap puede aplicarse tanto a filas como a columnas y define la separación entre las celdas adyacentes en la cuadrícula.

La sintaxis para utilizar gap en una cuadrícula es la siguiente:

```
.grid-container {  
  display: grid;  
  gap: valor;  
}
```

Donde valor puede ser un número seguido de una unidad de medida (por ejemplo, px, em, %, etc.).

row-gap y column-gap

En CSS Grid, row-gap y column-gap son propiedades individuales que se utilizan para establecer el espacio específicamente entre filas y columnas, respectivamente. Son variantes de la propiedad gap, que establece tanto el espacio entre filas como el espacio entre columnas de una cuadrícula.

row-gap:

Esta propiedad se utiliza para establecer el espacio entre filas en una cuadrícula. Define la separación vertical entre las celdas adyacentes en la cuadrícula en la dirección vertical (eje Y). Puedes especificar un valor numérico seguido de una unidad de medida (por ejemplo, píxeles, em, rem, porcentaje, etc.)

Ejemplo de uso:

```
.grid-container {  
  display: grid;  
  row-gap: 20px;  
  Establecer un espacio de 20px entre filas  
}
```

column-gap:

Esta propiedad se utiliza para establecer el espacio entre columnas en una cuadrícula. Define la separación horizontal entre las celdas adyacentes en la cuadrícula en la dirección horizontal (eje X). Al igual que con row-gap, puedes especificar un valor numérico con una unidad de medida.

Ejemplo de uso:

```
.grid-container {  
  display: grid;  
  column-gap: 20px;  
  Establecer un espacio de 20px entre columnas  
}
```

Ahora que ya sabemos que es CSS Grid a continuación veamos unos ejercicios: