GRIDS II

# Grids Mobile First

Antes de hablar de “Mobile-First” debemos hacer una referencia al llamado Diseño Responsive. Se refiere a la idea de que un sitio web debería mostrarse igual de bien en todo tipo de dispositivo, desde monitores de pantalla panorámica hasta teléfonos móviles. Es un enfoque para el diseño y desarrollo web que elimina la distinción entre la versión amigable para dispositivos móviles de un sitio web y su contraparte de escritorio. Con un diseño responsive ambos son lo mismo.

"Mobile First" significa crear el código primero para los dispositivos más pequeños que los usuarios probablemente tengan, como teléfonos o tabletas. Trabajar en el dispositivo más pequeño y luego acumular desde allí todo en el mismo código y el mismo proyecto, en lugar de uno nuevo para cada tamaño de pantalla.

Se recomienda:

👉 Primero trabajar el código para que se reproduzca perfectamente en un teléfono.

👉 Segundo, ajustar para que se ejecute en una tableta.

👉 Por último, trabajar en un dispositivo de escritorio.

Cualquier estilo dentro del siguiente Media Querie se ejecutará cuando el tamaño de la pantalla sea de al menos 768px de ancho -tablet portrait iPad Mini- pero no cuando el tamaño de la pantalla sea menor:

Estilos Mobile

@media screen and (min-width: 768px) {

.body {

background-color: #000000;

}

## Media Queries

👉 El diseño responsive se logra a través de "Media Queries" de CSS.

👉 Pensemos en las Media Queries como una forma de aplicar condicionales a las reglas de CSS.

👉 Estas últimas le dicen al navegador qué reglas debe ignorar o aplicar dependiendo del dispositivo del usuario.

## Break Point: categorías

| **Tamaño** | **Dispositivo** |
| --- | --- |
| 320px | Para dispositivos con pantallas pequeñas, como los teléfonos en modo vertical |
| 480px | Para dispositivos con pantallas pequeñas, como los teléfonos, en modo horizontal |
| 600px | Tabletas pequeñas, como el Amazon Kindle (600×800) y Barnes & Noble Nook (600×1024), en modo vertical |
| 768px y 1023px | Tabletas de diez pulgadas como el iPad (768×1024), en modo vertical |
| 1024px | Tabletas como el iPad (1024×768), en modo horizontal, así como algunas pantallas de ordenador portátil, netbook, y de escritorio |
| 1200px | Para pantallas panorámicas, principalmente portátiles y de escritorio |

## Estructura HTML paso a paso

1. Lo primero es **asignarle** a nuestro contenedor la propiedad de display: grid;
2. Luego, **número de columnas y filas que tendrá nuestra grilla**, y un espacio de separación.
3. **Definir** el área que ocupará cada caja de nuestro contenedor, primero le asignaremos un nombre y un color característico.
4. **Definir** cómo queremos que cada área sea acomodada en nuestro layout

## Tablet: siguiendo el ejemplo de grillas por áreas

👉 Para la versión tablet lo primero que hacemos es cambiar la disposición de las columnas de nuestro Grid.

👉 Luego cambiar la disposición de los ítems, este vez usando el recurso de grid-row y grid-column, que es el método corto de grid-row-start/end grid-column-start/end.

#grilla { display: grid; }

@media screen and (min-width: 768px) {

#grilla {

grid-template-columns: repeat(4, 1fr);

}

.border {

border: 4px solid black;

background-color: blue;

}  
}

## Desktop: siguiendo el ejemplo de grillas por áreas

👉 Cambiamos la disposición de la grilla.

👉 Y ahora una vez más cambiamos la disposición de los ítems

@media screen and (min-width: 1024px) {

#grilla {

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

}

.border {

border: 4px solid black;

background-color: green;

}  
}

## 

## Meta Viewpoint

Las páginas optimizadas para diferentes dispositivos deben incluir la etiqueta <meta> viewport en el encabezado del documento HTML. Una etiqueta <meta> viewport da al navegador las instrucciones sobre cómo controlar las dimensiones y el ajuste a escala de la página.

✓ Usa la etiqueta <meta> viewport para controlar el ancho y el ajuste de la ventana de visualización del navegador.

✓ Incluye width=device-width para hacer coincidir el ancho de la pantalla en píxeles independientes del dispositivo.

✓ Usa initial-scale=1 para establecer una relación de 1:1 entre los píxeles CSS y los píxeles independientes del dispositivo.

El uso del valor de width=device-width indica a la página que debe hacer coincidir el ancho de la pantalla en píxeles independientes del dispositivo. Esto permite que la página realice el reprocesamiento del contenido para adaptarlo a diferentes tamaños de pantalla.

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

