

{desafío}
latam_

Diseño responsivo _

Parte II



Orden de media queries

Desktop First

```
/* Estilos */  
@media screen and (max-width:  
1024px) {  
  // reglas CSS  
}  
  
@media screen and (max-width:  
768px) {  
  // Reglas CSS  
}
```

- Los estilos en esta técnica se construyen usando una serie de media queries que especifican el ancho máximo.
- Por lo general el diseño en dispositivos pequeños es simple.

Mobile First

```
/* Estilos */
@media screen and (min-width:
320px) {
  // reglas CSS
}

@media screen and (min-width:
768px) {
  // Reglas CSS
}
```

- Este método utiliza los media queries para ajustar el diseño en resoluciones más grandes, especificando el ancho mínimo que tendrá el primer dispositivo a afectar.

Breakpoints

```
/* Estilos */  
@media screen and (min-width:  
320px) {  
  // reglas CSS  
}
```

- El concepto de breakpoint se refiere al punto en el cual el dispositivo aplica el bloque de código contenido dentro de un *media query*.

Breakpoints de Bootstrap

Dispositivo	Tamaño
Dispositivos pequeños	<i>@media (min-width: 576px) { ... }</i>
Dispositivos medianos	<i>@media (min-width: 768px) { ... }</i>
Dispositivos grandes	<i>@media (min-width: 992px) { ... }</i>
Dispositivos extra grandes	<i>@media (min-width: 1200px) { ... }</i>

Componentes de un diseño responsivo: Grillas

Importancia de la grillas



- Las grillas cumplen una función primordial en el diseño responsivo, ya que su uso nos permitirá conocer cómo se moverá el flujo de los elementos y cómo se dispondrá el contenidos sin importar el tamaño de dispositivo a usar.

¿Cómo elegir una grilla?

Podemos elegir una grilla siguiendo alguno de estos principios:



- La grilla deberá solucionar problemas.
- La grilla será una componente de la experiencia de usuario.
- Mientras más simple sea la grilla más efectiva será.

CSS Grid

¿Qué es CSS Grid?



- CSS Grid es una tecnología CSS que nos ayudará a crear un sistema de grillas a modo de posicionar elementos en una interfaz de usuario.

Estructura base de CSS Grid

Para poder crear una grilla con esta tecnología debemos tener en cuenta la estructura usada por CSS Grid:

Contenedor

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

Ítems

Propiedades de contenedor

Display: grid

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

- Para definir la grilla debemos agregar al contenedor la propiedad display: grid.
- Esta propiedad definirá que el display del contenedor será grid.

Definir filas y columnas

```
.container {  
  display: grid;  
  grid-template-columns: 200px 200px;  
}
```

Con la propiedad `grid-template-columns` podremos definir el tamaño que queremos que tengan las columnas.

Con `grid-template-rows` definimos las filas.

Unidad de medida *fr*

```
.container {  
  display: grid;  
  grid-template-columns: 1fr 1fr;  
}
```

- *fr* es una unidad de medida específica de CSS grid que nos ayudará a **crear grillas fluidas**.
- Esta unidad nos permitirá definir la fracción de espacio que queremos usar dentro de la grilla.

Repitiendo tamaños

```
.container {  
  display: grid;  
  grid-template-columns: repeat(3,  
1fr);  
}
```

- La notación **repeat()** nos permitirá repetir todas o solamente una sección de nuestra filas o columnas.

Espacio entre los ítems

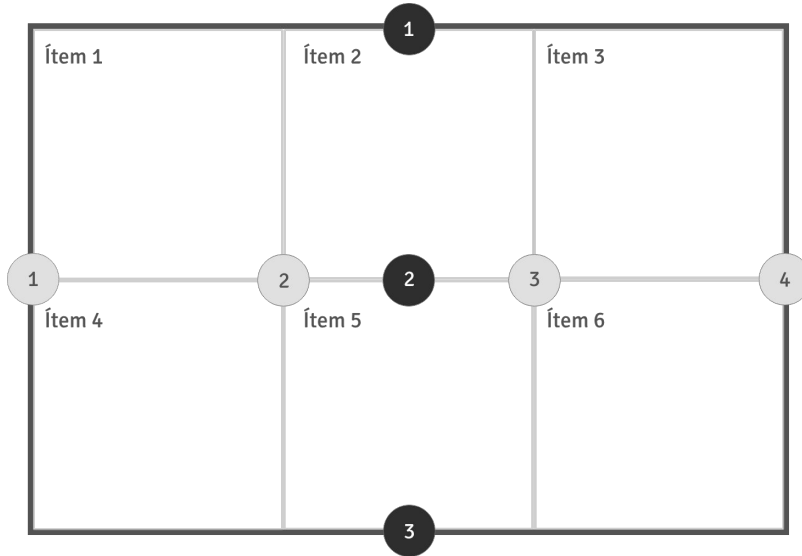
```
.container {  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  grid-gap: 20px;  
}
```

La propiedad ***grid-gap*** del contenedor nos permitirá definir el espacio que hay entre cada uno de los componentes de la grilla.

Propiedades de ítems

Líneas verticales en grilla

Para poder posicionar los ítems es importante conocer las líneas verticales:



- Las líneas verticales nos permitirán identificar dónde comienza y termina la grilla.
- El saberlo nos permitirá posicionar de manera más fácil los ítems de la grilla.

Posicionar ítems

```
.item-3 {  
  grid-column-start: 1;  
  grid-column-end: 3;  
}
```

- Si queremos definir la ubicación de un ítem dentro la grilla podremos hacerlo con `grid-column-start` o `grid-row-start` si se comienza desde el principio y `grid-column-end` o `grid-rows-end` si es desde el final de la línea vertical.
- Una versión corta de esta misma es `grid-column` o `grid-row`.

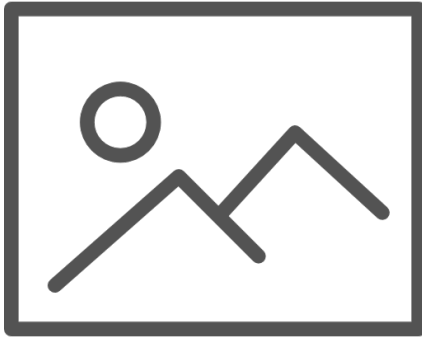
Posicionar en base a columnas

```
.item-3 {  
  grid-column: span 2;  
}
```

- Si queremos posicionar los elementos basados en la cantidad de líneas podremos usar un valor llamado ***span***.
- Este valor no permitirá especificar la cantidad de columnas que abarcar el elemento.

Componentes de un diseño responsivo: Imágenes responsivas

¿Qué son las imágenes responsivas?



- Las imágenes responsivas son tipos de imágenes los cuales usan escalan su tamaño a modo de fluir con el espacio disponible en el diseño.

Estrategias para transformar imágenes a responsivas

Uso de anchos y porcentajes

```
img {  
  width: 100%;  
  height: auto;  
}
```

```
img {  
  max-width: 100%;  
  height: auto;  
}
```

- Esta consta de agregar un ancho definido por porcentajes y un alto automático.
- Esto hace que tamaño de la imagen aumente o disminuya dependiendo del ancho del viewport.
- También es común encontrar una versión alternativa de esta estrategia definiendo el ancho como ancho máximo.

Background escalable

```
.hero-section__featured-image {  
  display: none;  
  background-image:  
url(../main-image.jpg);  
  background-repeat: no-repeat;  
  background-position: center;  
  // opciones: 100% 100%, cover o contain  
  background-size: cover;  
}
```

- Otra estrategia interesante a aplicar en nuestros proyectos es el uso de imágenes de fondo para escalarlas.
- Para hacerlo debemos agregar un tamaño de fondo con **porcentajes**, con valor cover o **contain**.

Uso de srcset

```
<picture>  
  <source media="(min-width: 1200px)" srcset="image@2x.jpg">  
    
</picture>
```

Esta estrategia permite agregar más de una imagen en una misma etiqueta haciendo que el navegador decida qué imagen mostrará en base a un tamaño detectado.

{desafío}
latam_

*Academia de
talentos digitales*

www.desafiolatam.com