# Conceptos Fundamentales del Grid de Bootstrap

**Contenedor (.container)**: Centraliza y proporciona un ancho máximo al contenido.

.container: ancho fijo dependiendo del breakpoint.

.container-fluid: ocupa el 100% del ancho.

**Fila (.row)**: Contiene columnas. Añade márgenes negativos y alinea contenido horizontalmente.

**Columna (.col)**: Elemento flexible dentro de .row. Bootstrap divide el espacio horizontal en **12 columnas** por fila.

## Breakpoints (Responsive)

Usa prefijos para ajustar el diseño en diferentes tamaños de pantalla:

|  |  |  |
| --- | --- | --- |
| **Breakpoint** | **Prefijo de Clase** | **Min-width** |
| xs | .col- | 0px |
| sm | .col-sm- | 576px |
| md | .col-md- | 768px |
| lg | .col-lg- | 992px |
| xl | . col-xl- | 1200px |
| xxl | .col-xxl- | 1400px |

## Tipos de Columnas

* .col: Auto-layout, divide el espacio restante en partes iguales.
* .col-{n}: Ocupa **n de 12 columnas**.
* .col-{breakpoint}-{n}: Responsive con tamaño específico.
* .col-auto: Se ajusta al contenido.

### Ejemplos de uso

*html*

*<!-- Tres columnas iguales -->*  
*<div class="container">*  
 *<div class="row">*  
 *<div class="col">1</div>*  
 *<div class="col">2</div>*  
 *<div class="col">3</div>*  
 *</div>*  
*</div>*  
----------------------------------------------------------------------------------------------------------------  
*<!-- Distribución responsive -->*  
*<div class="row">*  
 *<div class="col-sm-6 col-md-4">A</div>*  
 *<div class="col-sm-6 col-md-8">B</div>*  
*</div>*

Clases útiles

* .row-cols-{n}: Fuerza cantidad de columnas por fila.
* .gx-\* y .gy-\*: Controlan el **gutter** horizontal y vertical.
* .g-0: Sin gutters.
* .justify-content-\*, .align-items-\*: Controlan alineación en flexbox.
* **Anidamiento**: Puedes tener .row dentro de una .col para crear subgrillas.

## Sass (opcional para personalización avanzada)

Variables:

*$grid-columns: 12;*  
*$grid-gutter-width: 1.**5rem;*

Mixins:

*// Crea un contenedor para una serie de columnas*  
*@include make-row();*

# **Contenedores**

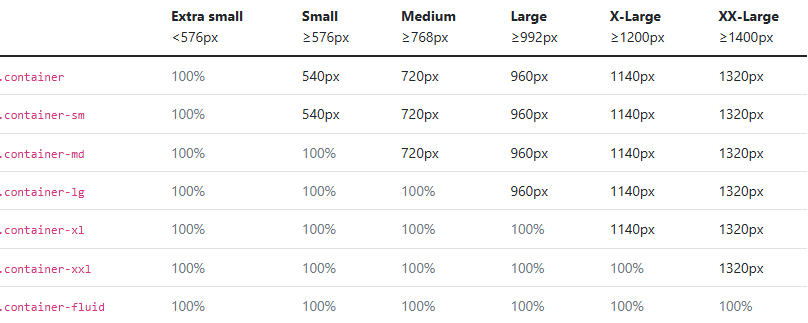
Los **contenedores** son elementos esenciales en Bootstrap. Su función principal es **encerrar, rellenar y alinear el contenido** dentro de un área visible del dispositivo (viewport). Estos elementos son imprescindibles cuando se utiliza el sistema de cuadrícula de Bootstrap.

## **Tipos de contenedores**

Bootstrap ofrece tres tipos principales de contenedores:

* .container: define un **ancho máximo (max-width)** que varía según el tamaño del viewport.
* .container-fluid: ocupa siempre el **100% del ancho disponible**, en todos los tamaños de pantalla.
* .container-{breakpoint}: tiene **ancho completo** hasta alcanzar un tamaño de pantalla específico (breakpoint), y luego adopta un **max-width** como .container.

### **Tabla comparativa de anchos máximos (max-width)**



## **Contenedor predeterminado**

El contenedor básico (.container) tiene un **ancho fijo y sensible** que se adapta a cada breakpoint.

*html*

*<div class="container">*  
 *<!-- Tu contenido aquí -->*  
*</div>*

## **Contenedores responsivos**

Los **contenedores responsivos** permiten usar clases que tienen **100% de ancho hasta cierto breakpoint**, momento en el cual se aplican valores de max-width específicos para pantallas más grandes.

*html*

*<div class="container-sm">100% de ancho hasta el breakpoint "sm"</div>*  
*<div class="container-md">100% de ancho hasta "md"</div>*  
*<div class="container-lg">100% de ancho hasta "lg"</div>*  
*<div class="container-xl">100% de ancho hasta "xl"</div>*  
*<div class="container-xxl">100% de ancho hasta "xxl"</div>*

## **Contenedores fluidos**

Utiliza .container-fluid cuando necesites que el contenido **abarque el ancho completo del viewport**, sin importar el tamaño de la pantalla.

*html*

*<div class="container-fluid">*  
 *<!-- Contenido de ancho completo -->*  
*</div>*

## **Personalización con Sass**

Bootstrap personalizamos los tamaños máximos de los contenedores a través del mapa $container-max-widths, definido en \_variables.scss:

*$container-max-widths: (*  
 *sm: 540px,*  
 *md: 720px,*  
 *lg: 960px,*  
 *xl: 1140px,*  
 *xxl: 1320px );*

// Definición del mixin  
*@mixin make-container($padding-x: $container-padding-x) {*  
 *width: 100%;*  
 *padding-right: $padding-x;*  
 *padding-left: $padding-x;*  
 *margin-right: auto;*  
 *margin-left: auto; }*