

Bootstrap permite un máximo de 12 columnas en toda la página  
Las columnas se reorganizan en función del tamaño de la pantalla

lapso 1	lapso 1	lapso 1	lapso 1	lapso 1	lapso 1	lapso 1	lapso 1	lapso 1	lapso 1	lapso 1	lapso 1
lapso 4				lapso 4				lapso 4			
lapso 4				lapso 8							
lapso 6						lapso 6					
span 12											

## Clases de cuadrícula

Son 5:

- .col- (Dispositivos adicionales pequeñas - ancho de la pantalla de menos de 576px)
- .col-sm- (Pequeños dispositivos - ancho de la pantalla igual o mayor que 576px)
- .col-md- (Dispositivos medio - ancho de la pantalla igual o mayor que 768px)
- .col-lg- (Dispositivos de gran tamaño - ancho de la pantalla igual o mayor que 992px)
- .col-xl- (Dispositivos XLarge - ancho de la pantalla igual o mayor que 1200px)

Algunas reglas del sistema 4 rejilla Bootstrap:

- Filas deben ser colocados dentro de un .container(ancho fijo) o .container-fluid(ancho completo) para la alineación y el relleno adecuado
- Utilice filas para crear grupos de columnas horizontales
- El contenido debe ser colocado dentro de las columnas, y sólo las columnas puede ser hijos inmediatos de filas
- Clases predefinidas son .row y .col-sm-4 están disponibles para hacer rápidamente diseños de cuadrícula
- Columnas crean canalones (huecos entre contenido de la columna) a través de relleno. Que el relleno se compensa en filas para la primera y última columna a través de margen negativo en .rows
- columnas de la cuadrícula se crean especificando el número de 12 columnas disponibles que desea abarcar. Por ejemplo, tres columnas iguales usarían tres.col-sm-4
- ancho de las columnas están en porcentaje, por lo que siempre son fluidos y dimensionados en relación a su elemento padre
- La mayor **diferencia entre Bootstrap 3 y 4 Bootstrap** es que Bootstrap 4 utiliza FlexBox en lugar de float. Una gran ventaja con FlexBox es que las columnas de la cuadrícula sin una anchura especificada se layout automáticamente columnas como la igualdad de anchura. Ejemplo: tres elementos con .col-smcada uno será automáticamente 33,33% amplia desde el pequeño punto de interrupción y hacia arriba.

## FLEXBOX

Un diseño FlexBox consiste en un elemento padre, con uno o más elementos secundarios.

Propiedades:

- Flex-direccion: define en qué dirección quiere que el contenedor apile los artículos
  - Column: De arriba abajo
  - Column-reverse: de abajo a arriba
  - Row: para apilar los elementos de izquierda a derecha
  - row-reverse: para apilar los elementos de una fila de derecha a izquierda
- La propiedad flex-wrap
  - Wrap. Para mostrar los valores en varias filas
  - Nowrap: Para mostrar los valores en una fila

- wrap-reverse: varias filas empezando desde abajo
- flex-flow: propiedad abreviada de flex-direction y flex-wrap
- justify-content Property: alineacion
  - justify-content: center
  - justify-content: flex-start
  - justify-content: flex-end;
  - align-items: stretch;
  - align-items: baseline;
  - justify-content: space-around;
  - justify-content: space-between
  - align-items: center;

display Especifica el tipo de caja utilizada para un elemento de HTML

flex-direction Especifica la dirección de los elementos flexibles dentro de un contenedor flexible

justify-content Alinea horizontalmente los elementos flexibles cuando los elementos no usan todo el espacio disponible en el eje principal

align-items Alinea verticalmente los elementos flexibles cuando los elementos no usan todo el espacio disponible en el eje transversal

flex-wrap Especifica si los elementos flexibles deben envolverse o no, si no hay espacio suficiente para ellos en una línea flexible

align-content Modifica el comportamiento de la propiedad flex-wrap. Es similar a alinear elementos, pero en lugar de alinear elementos flexibles, alinea líneas flexibles

flex-flow Una propiedad taquigráfica para la flexión y flex-wrap

order Especifica el orden de un artículo flexible en relación con el resto de los artículos flexibles dentro del mismo contenedor

align-self Usado en artículos flexibles Anula la propiedad de elementos de alineación del contenedor

# Ejemplo 1 de cuadrícula Bootstrap: apilado a horizontal

El siguiente ejemplo muestra un diseño simple de dos columnas "apiladas a horizontales", lo que significa que dará lugar a una división del 50% / 50% en todas las pantallas, excepto en pantallas extrapequeñas, que se acumularán automáticamente (100%):

## Grid

This example demonstrates a 50%/50% split on small, medium and large devices. On extra small devices, it will stack (100% width).

Resize the browser window to see the effect.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto.

# Ejemplo 2 de Bootstrap Grid: dispositivos pequeños

	Extra small	Small	Medium	Large
Class prefix	<code>.col-xs</code>	<code>.col-sm</code>	<code>.col-md</code>	<code>.col-lg</code>
Screen width	<576px	<b>&gt;=768px</b>	>=992px	>=1200px

Supongamos que tenemos un diseño simple con dos columnas. Queremos que las columnas se dividan en un 25% / 75% para dispositivos pequeños.

**Consejo:** Los dispositivos pequeños se definen por tener un ancho de pantalla de **768 a 991 píxeles**.

Para dispositivos pequeños usaremos las `.col-sm-*` clases.

Añadiremos las siguientes clases a nuestras dos columnas:

Ahora Bootstrap va a decir "en el tamaño pequeño, busca clases con `-sm-` en ellas y úsalas".

El siguiente ejemplo dará como resultado una división del 25% / 75% en dispositivos pequeños (y medianos y grandes). En dispositivos extra pequeños, se acumulará automáticamente (100%):

`col-sm-3`

`col-sm-9`

## Ejemplo 3: Tres columnas iguales

El siguiente ejemplo muestra cómo obtener tres columnas de ancho igual comenzando en tabletas y escalando a escritorios grandes. En teléfonos móviles, las columnas se acumularán automáticamente:

## Ejemplo 4: Tres columnas desiguales

El siguiente ejemplo muestra cómo obtener tres columnas de varios ancho comenzando en tabletas y escalando a escritorios grandes:

## Ejemplo 5: Dos columnas desiguales

El siguiente ejemplo muestra cómo obtener dos columnas de ancho variable comenzando en tabletas y escalando a escritorios grandes:

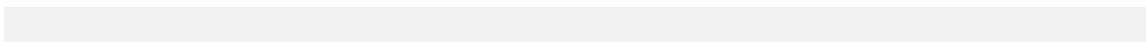
## Ejemplo 6: Dos columnas con dos columnas anidadas

El siguiente ejemplo muestra cómo obtener dos columnas comenzando en tabletas y escalando a escritorios grandes, con otras dos columnas (anchos iguales) dentro de la columna más grande (en teléfonos móviles, estas columnas y sus columnas anidadas se apilarán):

## Ejemplo 7: Mixto: móvil y escritorio

El sistema de grillas de Bootstrap tiene cuatro clases: xs (teléfonos), sm (tabletas), md (computadoras de escritorio) y lg (computadoras de escritorio más grandes). Las clases se pueden combinar para crear diseños más dinámicos y flexibles.

**Consejo:** cada clase aumenta, por lo que si desea establecer los mismos anchos para xs y sm, solo necesita especificar xs.



## Ejemplo 8: Mixto: móvil, tableta y escritorio

## Ejemplo 9: Flotadores claros

Flotadores claros (con la `.clearfix` clase) en puntos de interrupción específicos para evitar envoltorios extraños con contenido desigual:

## Ejemplo 10: Columnas de desplazamiento

Mueva las columnas hacia la derecha usando `.col-md-offset-*` clases. Estas clases aumentan el margen izquierdo de una columna por \* columnas:

## Ejemplo 11: Empujar y tirar - Cambiar orden de columna

Cambia el orden de las columnas de la grilla con `.col-md-push-*` y las `.col-md-pull-*` clases: