

Tema 8. Frameworks CSS: Bootstrap

Experiencia de Usuario en la Web

Índice

- ▶ Introducción
- ▶ Primeros pasos con Bootstrap
 - Descarga e instalación
 - Plantilla básica
 - Compatibilidad
 - Personalización
 - Documentación
- ▶ Diseño de la página
- ▶ Imágenes y flotantes
- ▶ Ocultar y mostrar contenido

Nota: En el momento de elaborar esta documentación la última versión disponible de Bootstrap era la versión v4.0.0-beta. Aunque se trata de una versión beta, se ha preferido utilizarla previendo que a lo largo del curso aparezca la primera versión estable. Es posible que algunas características de bootstrap sufran cambios en los próximos meses.

Introducción

Introducción

- ▶ Un framework css es un conjunto de herramientas (hojas de estilo, scripts, etc.) y prácticas que facilitan las tareas repetitivas para trabajar con css.
- ▶ Aporta una serie de scripts y hojas de estilo que se aplican en forma de clases al código html.
- ▶ Facilita la creación de *layouts* basados en *responsive design* mediante retículas.
- ▶ También incorpora elementos avanzados para incluir en la página.
 - Menús desplegados, carruseles, migas de pan, controles de formulario rediseñados, control de la presentación, etc.
- ▶ Además, incorporan herramientas que permiten anular los estilos por defecto de los navegadores, facilitando la homogeneidad en la visualización del contenido en distintos navegadores.

Introducción (II)

- ▶ La utilización de frameworks css también presenta inconvenientes:
 - Homogeneiza la presentación, haciendo que todos los sitios web tengan un aspecto similar.
 - Aunque es posible modificar el aspecto de los distintos elementos de diseño, eso implica recodificar los estilos del propio framework, perdiendo la ventaja de la simplicidad y el ahorro de tiempo.
 - Obliga a utilizar la semántica propia del framework.
 - Si se está habituado a utilizar una nomenclatura para clases y selectores css, puede ser un inconveniente tener que cambiarla.
 - Esto puede ser una dificultad importante si se está trabajando con un equipo de colaboradores.
 - Añade código, muchas veces no utilizado, en nuestro sitio web.
 - Es la desventaja principal.
 - Aunque se posible limitar el número de clases que se importa e incluso limpiar el código css no utilizado, esto implica tiempo y tener un conocimiento profundo de css y del propio framework.

Introducción (III)

- ▶ Algunos ejemplos de frameworks css...
 - Foundation for sites (foundation.zurb.com/sites.html)
 - UIKit (getuikit.com/)
 - Bulma (bulma.io/)
 - Pure css (purecss.io/)
 - Skeleton (getskeleton.com/)
 - Simple Grid (thisisdallas.github.io/Simple-Grid/)
 - 960 Grid System (960.gs/)
 - Unsemantic (unsemantic.com/)
- ▶ Bootstrap es un framework css que facilita la tarea de diseño de una página web.
 - Aunque no es el más ligero, en la actualidad es el más popular y cuenta con una extensa comunidad de desarrolladores, temas, plantillas y plugins.

Primeros pasos con Bootstrap

Primeros pasos con Bootstrap: descarga e instalación

- ▶ Desde <http://getbootstrap.com/> podemos descargar:
 - css compilado.
 - Incluye css compilados y las versiones comprimidas.
 - Scripts compilados y las versiones mínimas de Javascript para los plugin.
 - Código fuente en formato Sass (Syntactically Awesome Stylesheets, sass-lang.com/):
 - Incluye los archivos Sass, Javascript y la documentación.
 - Facilita la integración de Bootstrap con otros proyectos basados en Sass (Ruby on Rails, Compass, etc.).
 - Requiere un compilador de Sass.
 - Las versiones anteriores también distribuían el código en formato Less.
 - Less (lesscss.org/) es un preprocesador css que permite crear css dinámico.
- ▶ En la actualidad la última versión disponible es la versión 4.

Primeros pasos con Bootstrap: descarga e instalación (II)

- ▶ También es posible acceder al framework mediante CDN (Content Delivery Network).
 - Los archivos compilados están almacenados en el sitio [bootstrapcdn.com](https://maxcdn.bootstrapcdn.com).

```
<link rel="stylesheet"
      href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css"
      integrity="sha384-Y6pD6FV/Vv2HJnA6t+vsIU6fwYXjCFTcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M"
      crossorigin="anonymous">
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/js/bootstrap.min.js"
        integrity="sha384-h0AbiXch4ZDo7tp9hKZ4TshBi047NrKGL03SEJAg45jXxnGIIfYzk4Si90RDIqNm1"
        crossorigin="anonymous"></script>
```

- Para la aplicación de algunos elementos y plugin, también es necesario hacer las llamadas a `jquery.js` (una biblioteca de Javascript) y `popper.js` (una biblioteca que permite la creación de etiquetas con información sobre un elemento web).

```
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"
        integrity="sha384-KJ3o2DKtIkVYIK3UENzmM7KCKRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
        crossorigin="anonymous"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js"
        integrity="sha384-b/U6ypiBEHpOf/4+1nzFpr53nxSS+GLCKfwBdFNTxtclqqenISfwAzpKaMNFNmj4"
        crossorigin="anonymous"></script>
```

Primeros pasos con Bootstrap: descarga e instalación (III)

- ▶ La versión compilada se distribuye en dos carpetas.
 - Carpeta css:
 - Contiene la hoja de estilos principal (bootstrap.css) y hojas de estilo adicionales que contienen sólo los estilos correspondientes a la rejilla (bootstrap-grid.css) y para la inicialización de las hojas de estilo que tienen por defecto los navegadores (bootstrap-reboot.css).
 - Incluye las versiones comprimidas (sin saltos de línea ni retornos de carro) de las tres hojas de estilo (bootstrap.min.css, bootstrap-grid.min.css y bootstrap-reboot.min.css)
 - La versión de explotación de nuestro sitio sólo debería estas versiones minimizadas.
 - Se incluyen además archivos .map que se utilizan en algunas herramientas de desarrollo presentes en los navegadores.
 - Carpeta js:
 - Contiene los scripts bootstrap.js y bootstrap.min.js.
 - Son plugins para algunas funciones de bootstrap (transiciones, menús desplegados, navegación con pestañas, etc.).

Primeros pasos con Bootstrap: descarga e instalación (IV)

- ▶ Los plugins requieren la existencia de JQuery.
 - Se puede descargar desde jquery.com (jquery.com/download/).
 - Se puede utilizar mediante CDN incluyendo la llamada al sitio code.jquery.com/.
- ▶ Además, los tooltips y otros mensajes emergentes dependen de popper.js (github.com/FezVrasta/popper.js).
 - Se puede hacer la llamada al script de forma local o mediante CDN.

Primeros pasos con Bootstrap: plantilla básica

- ▶ La página debe estar hecha en html5.
- ▶ Para poder adaptar el contenido a dispositivos móviles necesita utilizar la etiqueta *meta* para el *viewport*.
- ▶ Necesita incluir la referencia a la hoja de estilos de Bootstrap.
 - Debe estar antes de nuestras hojas de estilo o estilos incrustados.
- ▶ Para utilizar los plugins hay que hacer las llamadas a jquery.js, popper.js y bootstrap.js.
 - Se colocan al final para optimizar la carga de código.

Primeros pasos con Bootstrap: plantilla básica (II)

```
<!doctype html>
<html lang="es">
<head>
  <title>Plantilla bootstrap 4.0 beta</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- Llamada al nucleo de bootstrap-->
  <link rel="stylesheet" href="css/bootstrap.min.css">
  <!-- Aquí pueden ir las hojas de estilo propias -->
</head>
<body>
  <h1>Plantilla básica de bootstrap</h1>

  <!-- Aquí se colocan los scripts para que la página se cargue más rápido -->
  <script src="js/jquery-3.2.1.min.js"></script>
  <script src="js/popper.js"></script>
  <script src="js/bootstrap.min.js"></script>
</body>
</html>
```

Primeros pasos con Bootstrap: plantilla básica (III)

- ▶ La plantilla anterior toma las hojas de estilo y los scripts de un almacenamiento local.
- ▶ También se pueden tomar de un repositorio.

```
<!doctype html>
<html lang="es">
<head>
  <title>Plantilla bootstrap 4.0 beta</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- Llamada al nucleo de bootstrap-->
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css"
    integrity="sha384-/Y6pD6FV/Vv2HJnA6t+vs1U6fwYXjCFtcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M"
    crossorigin="anonymous">
  <!-- Aquí pueden ir las hojas de estilo propias -->
</head>
<body>
  <h1>Plantilla básica de bootstrap</h1>
  <!-- Aquí se colocan los scripts para que la página se cargue más rápido -->
  <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"
    integrity="sha384-KJ3o2DKtIkvYIK3UEZnmM7KCRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
    crossorigin="anonymous"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js"
    integrity="sha384-b/U6ypiBEHpOf/4+1nzFpr53nxSS+GLCKfwbDfNtXtclqqenISfwAzpKaMNFNmj4"
    crossorigin="anonymous"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/js/bootstrap.min.js"
    integrity="sha384-h0AbiXch4ZDo7tp9hKZ4TsHbi047NrKGL03SEJAg45jXxnGIfYzk4Si90RDIqNm1"
    crossorigin="anonymous"></script>
</body>
</html>
```

Primeros pasos con Bootstrap: compatibilidad

- ▶ Bootstrap es compatible con la mayoría de las últimas versiones de los navegadores de escritorio y móviles.
- ▶ No es compatible con:
 - Opera para Android e iOS.
 - Internet Explorer con versiones anteriores a la 10.
- ▶ Para versiones de IE anteriores a la 10, hay que utilizar Bootstrap 3.
 - Nota: en este tema se ha utilizado la versión v4.0.0-beta, la última disponible a la hora de elaborarla documentación. Es posible que en futuras revisiones se incluya un soporte a versiones anteriores de IE.
- ▶ Bootstrap utiliza html5 y css3, lo que implica que puede que puede que no funcione de forma correcta en los navegadores que no lo soporten.
 - En caniuse.com/ se puede comprobar qué características de html5 y css3 soportan los distintos navegadores.

Primeros pasos con Bootstrap: compatibilidad (II)

- ▶ Se puede buscar cierta compatibilidad utilizando algunos scripts de Javascript.
 - **html5shiv** (github.com/aFarkas/html5shiv) es una biblioteca Javascript que permite reconocer la existencia de los nuevos elementos html (article, aside, header, footer, etc.)
 - **modernizr** (modernizr.com/) es una biblioteca Javascript que permite determinar si nuestro navegador soporta características de html5 y css.
 - Puede incluir también html5shiv.
 - **respond** (github.com/scottjehl/Respond) es un pollyfill que permite incluir *media queries* en navegadores que no lo permiten.
 - Fundamentalmente es aplicable a IE 6-8.

Primeros pasos con Bootstrap: personalización

- ▶ Los estilos de Bootstrap se pueden personalizar sobrescribiendo el archivo css.
- ▶ El archivo `bootstrap.css` presenta las clases en formato legible.
 - `bootstrap.min.css` es más difícil de leer.
- ▶ Para modificar una clase habría que:
 - Buscar el selector a modificar en `bootstrap.css` (es más fácil).
 - Copiar el selector en nuestra hoja de estilos o en el elemento *style* de la página.
 - Añadir o cambiar las propiedades.

Primeros pasos con Bootstrap: personalización

- ▶ Por ejemplo, en el ccs de Bootstrap existe el selector.
 - `h1,h2,h3,h4,h5,h6,.h1,.h2,.h3,.h4,.h5,.h6{font-family:inherit;font-weight:500;line-height:1.1;color:inherit}`
 - Las cabeceras no están en negrita (`font-weight:500` indica un grosor medio).
- ▶ La regla de estilo:
`h1,h2,h3,h4,h5,h6,.h1,.h2,.h3,.h4,.h5,.h6{
font-weight: bold;
}`
 - Haría que los encabezamientos aparecieran en negrita.
- ▶ La regla de estilo se debería colocar:
 - En un archivo css referenciado después de la llamada a los estilos de Bootstrap.
 - Dentro de una etiqueta `<style>`
 - El mantenimiento sería más difícil.
 - Las nuevas propiedades del selector sobrescriben las anteriores.

Primeros pasos con Bootstrap: personalización (II)

```
<!doctype html>
<html lang="es">
<head>
  <title>Plantilla bootstrap 4.0 beta</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- Llamada al nucleo de bootstrap-->
  <link rel="stylesheet" href="css/bootstrap.min.css">
  <!-- Aquí pueden ir las hojas de estilo propias -->
  <style>
    h1,h2,h3,h4,h5,h6,.h1,.h2,.h3,.h4,.h5,.h6{
      font-weight: bold;
    }
  </style>
  <!-- También podría incluirse la llamada a una hoja de estilo personalizada
       que incluyera la regla de estilo
       <link rel="stylesheet" href="css/miHojaDeEstilos.css">
  -->
</head>
...
```

- ▶ También es posible personalizar Bootstrap modificando las variables Sass y generando de nuevo la hoja de estilo.
 - Nota: Las versiones anteriores de Bootstrap incluían una utilidad que permitía seleccionar los elementos a incluir en bootstrap.css y personalizar las variables. Posiblemente la versión 4 definitiva también incluya esa utilidad.

Primeros pasos con Bootstrap: documentación

- ▶ En este tema nos vamos a ocupar fundamentalmente de los aspectos de Bootstrap que tiene que ver con el *layout* de la página y el *diseño responsive*.
- ▶ Para el resto de temas y componentes la documentación oficial es:
 - Para los elementos de contenido (listas, tablas, tipografía, imágenes, etc.): getbootstrap.com/docs/4.0/content/
 - Para los componentes de la interfaz (ventanas de alerta, botones, listas desplegadas, formularios, barras de navegación, etc.): getbootstrap.com/docs/4.0/components
 - Utilidades (alineación de texto, visibilidad, display, accesibilidad): getbootstrap.com/docs/4.0/utilities/

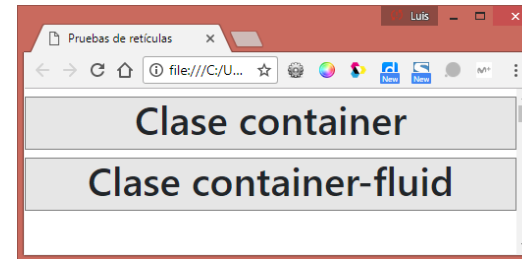
Diseño de la página

Diseño de la página

- ▶ Una de las ventajas principales de un framework css es la facilidad para realizar un diseño atractivo de la estructura de la página (layout) con características *responsive*.
- ▶ La manera de utilizar Bootstrap en una página web es asignando alguna de las clases definidas en la hoja de estilos a los distintos elementos estructurales y semánticos del código html de la página.
 - Para la estructura de la página, lo habitual es crear secciones `div` para cada elemento de la estructura y asignar una o más clases de Bootstrap.
 - También se pueden utilizar los propios elementos html para incluir las clases de Bootstrap.
- ▶ En el caso del diseño del *layout* de la página, los elementos de nivel superior son los contenedores.
- ▶ Los elementos que incluyamos dentro del contenedor se adaptarán a su ancho.
- ▶ Existen dos tipos de contenedores:
 - Clase `.container`: crea un contenedor con un ancho fijo que dependerá del ancho del *viewport* del navegador.
 - Clase `.container-fluid`: crea un contenedor de estilo fluido que siempre ocupará el ancho total del navegador.
- ▶ Aunque normalmente no es necesario, es posible anidar contenedores.

Diseño de la página (II)

```
<div class="container">
  <h1>Clase container</h1>
</div>
<div class="container-fluid">
  <h1>Clase container-fluid</h1>
</div>
```



Diseño de la página (III)

- ▶ Bootstrap utiliza una filosofía de diseño basada en el diseño *mobile first*.
- ▶ Los puntos de ruptura de las *media queries* se basan en la anchura mínima de los *viewport* de los navegadores.
 - Los estilos sin *media queries* son los que se utilizarían en dispositivos pequeños.
- ▶ Aunque es posible cambiarlos, estos son los puntos de ruptura principales establecidos.

```
// Dispositivos muy pequeños(teléfonos en modo retrato, menos de 576px)
// No requieren media queries. Son los estilos por omisión de bootstrap

// Dispositivos pequeños(teléfonos en modo horizontal, >= 576px)
@media (min-width: 576px) { ... }

// Dispositivos de tamaño medio(tablets, >=768px)
@media (min-width: 768px) { ... }

// Dispositivos grandes (ordenadores de escritorio, >= 992px)
@media (min-width: 992px) { ... }

// Dispositivos muy grandes (ordenadores con pantalla grande, >= 1200px)
@media (min-width: 1200px) { ... }
```


Diseño de la página (IV)

- En ocasiones también se utilizan *media queries* para dispositivos mayores de un rango determinado o dispositivos que estén entre dos rangos.

```
// Dispositivos muy pequeños(teléfonos en modo retrato, menos de 576px)
@media (max-width: 575px) { ... }
// Dispositivos pequeños(teléfonos en modo horizontal, < 768x)
@media (max-width: 767px) { ... }
// Dispositivos de tamaño medio(tablets, <992px)
@media (max-width: 991px) { ... }
// Dispositivos grandes (ordenadores de escritorio, <1200px)
@media (max-width: 1199px) { ... }
// Dispositivos muy grandes (ordenadores con pantalla grande, >= 1200px)
// No precisan media queries ya que no hay límite superior

// Dispositivos muy pequeños(teléfonos en modo retrato, menos de 576px)
@media (max-width: 575px) { ... }
// Dispositivos pequeños(teléfonos en modo horizontal, >=576px)
@media (min-width: 576px) and (max-width: 767px) { ... }
// Dispositivos de tamaño medio(tablets, >=768px)
@media (min-width: 768px) and (max-width: 991px) { ... }
// Dispositivos grandes (ordenadores de escritorio, >=992px)
@media (min-width: 992px) and (max-width: 1199px) { ... }
// Dispositivos muy grandes (ordenadores con pantalla grande, >= 1200px)
@media (min-width: 1200px) { ... }
```

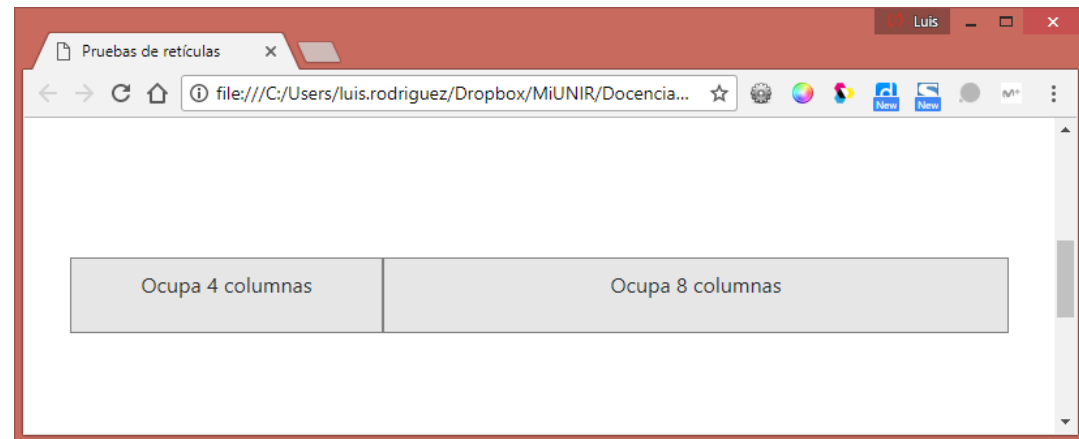
El sistema de retícula

- ▶ El sistema de retícula de bootstrap utiliza un layout de 12 columnas.
- ▶ Para crear las columnas de contenido se utilizan filas: contenedores de la clase `.row`.
 - Cada contenedor de la clase `.row` divide su espacio en 12 columnas separadas por *gutters* (en tipografía se denomina «calle», espacio entre columnas).
 - La anchura de cada columna depende de la anchura del contenedor.
 - Deben estar en un elemento de la clase `.container` o `.container-fluid`.
 - Solo se utilizan como contenedores de columnas.
 - El único elemento hijo directo de una fila solo puede ser una columna.
 - El contenido solo se coloca en las columnas, nunca en las filas.

El sistema de retícula (II)

- ▶ Las columnas (clases `.col-*`) de la retícula definen la anchura del bloque, especificando cuántas de las 12 columnas de la fila ocupan.
 - Si, por ejemplo, se quiere dividir una fila en una sección que ocupe 4 de las 12 columnas de la retícula y otra que ocupe 8 en un dispositivo mediano (aproximadamente de 1024 píxeles), se utilizarán las clases `.col-md-4` y `.col-md-8`.
 - md indica que es para dispositivos medianos.
 - El 4 y el 8 indica que una columna ocupa 4 de las 12 columnas en las que se divide cada fila y la otra 8.

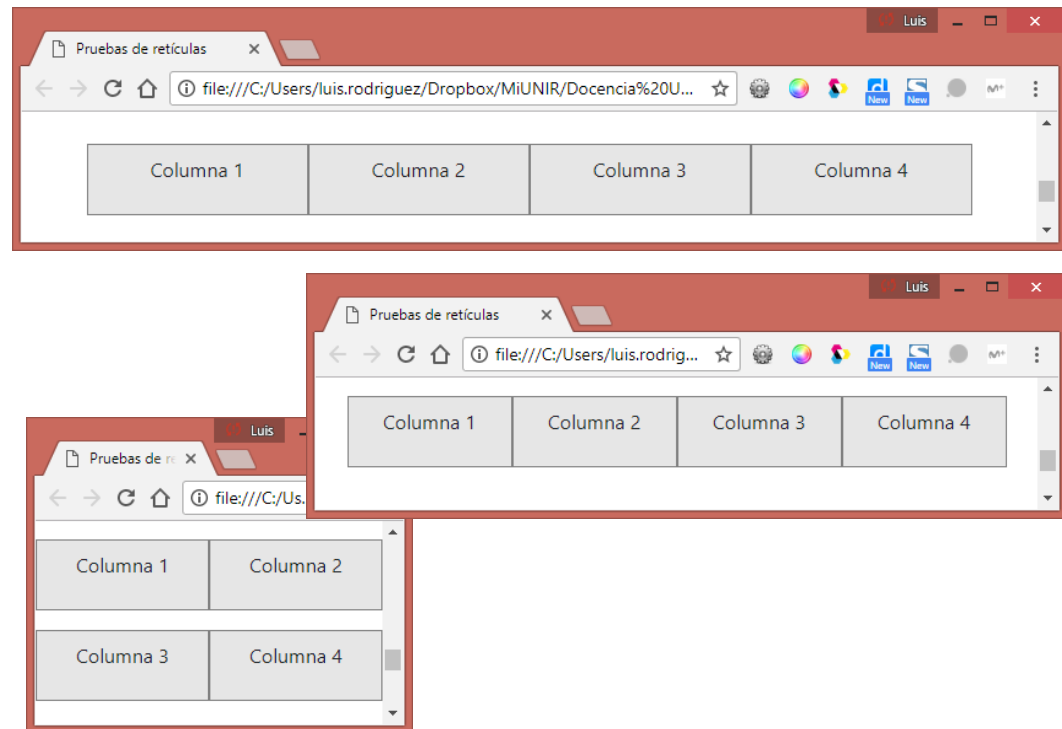
```
<div class="row">
  <div class="col-md-4">
    <p>Ocupa 4 columnas</p>
  </div>
  <div class="col-md-8">
    <p>Ocupa 8 columnas</p>
  </div>
</div>
```



El sistema de retícula (III)

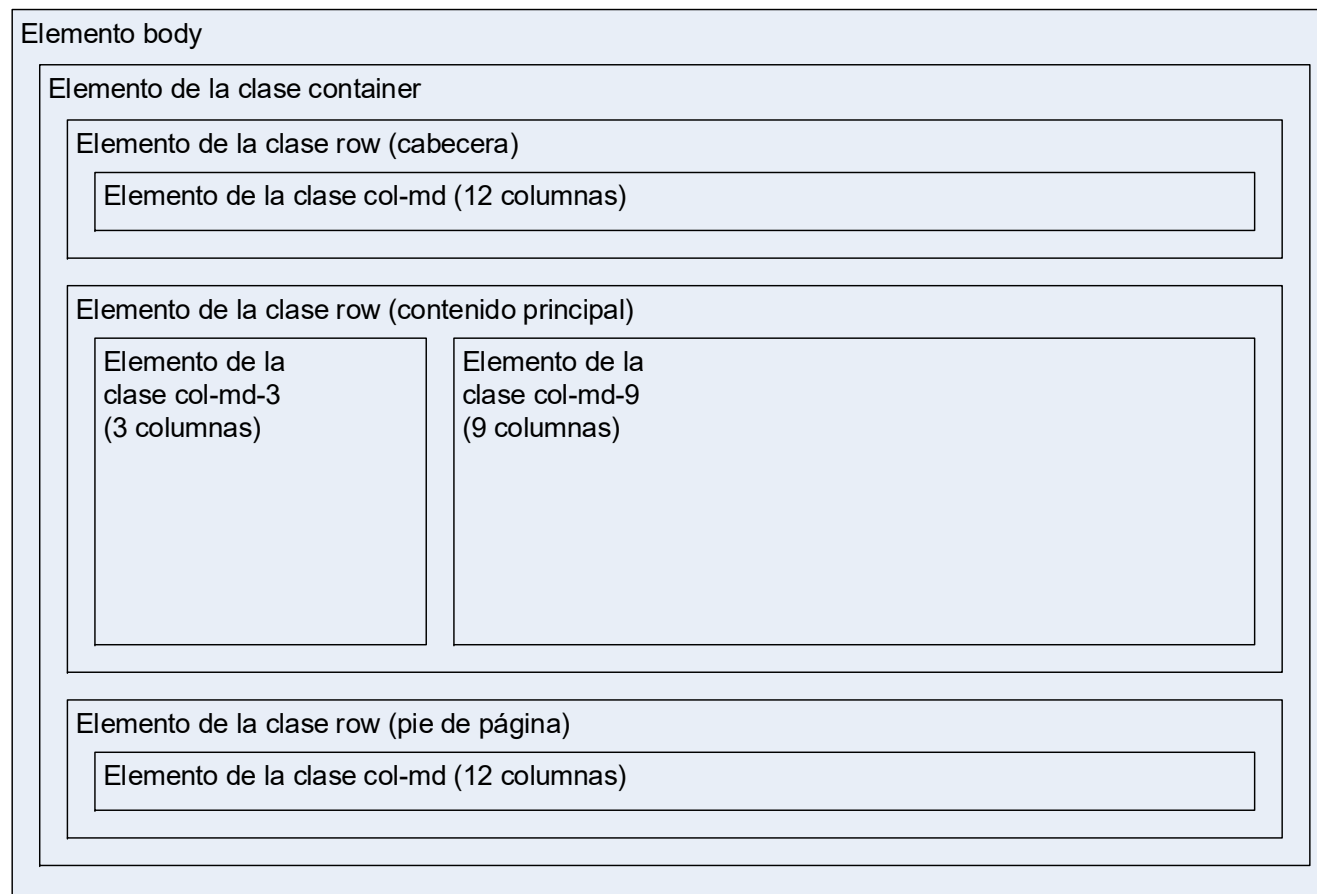
- ▶ El sistema de retícula utiliza el modelo de cajas flexibles de css 3.
 - Si no se especifica el ancho en columnas de las clases `.col-*`, el ancho de cada columna se distribuirá a partes iguales entre el número de columnas que se incluye en el contenedor `.row`.
 - El ancho mínimo también dependerá del contenido de la columna.

```
<div class="container">
  <div class="row">
    <div class="col">
      <p>Columna 1</p>
    </div>
    <div class="col">
      <p>Columna 2</p>
    </div>
    <div class="col">
      <p>Columna 3</p>
    </div>
    <div class="col">
      <p>Columna 4</p>
    </div>
  </div>
</div>
```



El sistema de retícula (IV)

- Una estructura típica de una página podría ser...



Clases para las columnas

- ▶ Bootstrap puede tener 5 tipos de retícula en el diseño de la página web que se corresponden con los 5 puntos de ruptura de las *media queries*: dispositivos muy pequeños, dispositivos pequeños, dispositivos medianos, grandes y muy grandes.
 - A cada uno de ellos le corresponde una clase de columna distinta.
- ▶ En general el nombre de las clases referentes a las columnas tiene el formato `.col-*-n`.
 - * puede ser:
 - sm, para dispositivos pequeños, tipo smartphone en modo horizontal
 - md, para dispositivos medianos, tipo tableta.
 - lg, para dispositivos grandes, tipo ordenador de escritorio.
 - xl, para dispositivos de escritorio con pantallas grandes.
 - Si no se incluye este elemento, se utilizará para dispositivos muy pequeños, como smartphone en modo retrato.
 - n es un número entre 1 y 12 que indica el número de columnas que ocupará el bloque.
 - Si no se incluye este número el espacio de la fila se distribuirá de forma equitativa entre el número de columnas dentro del propio contenedor.
- ▶ Los puntos de ruptura se corresponden a las anchuras mínimas, por lo que, por ejemplo, la clase `.col-md` se aplicará en todos los dispositivos medianos, grandes y muy grandes, siempre y cuando no se especifique también una clase para un dispositivo de mayor tamaño.

Características de las columnas

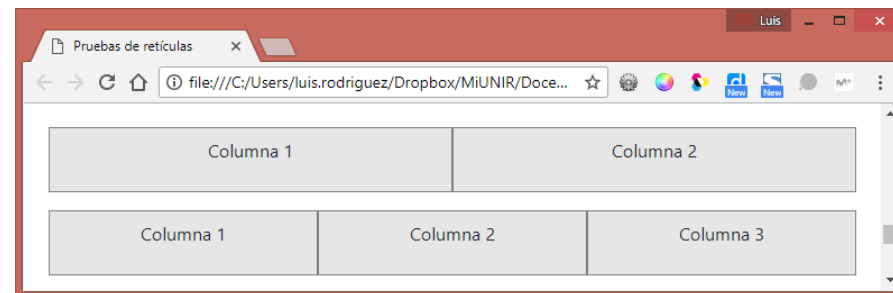
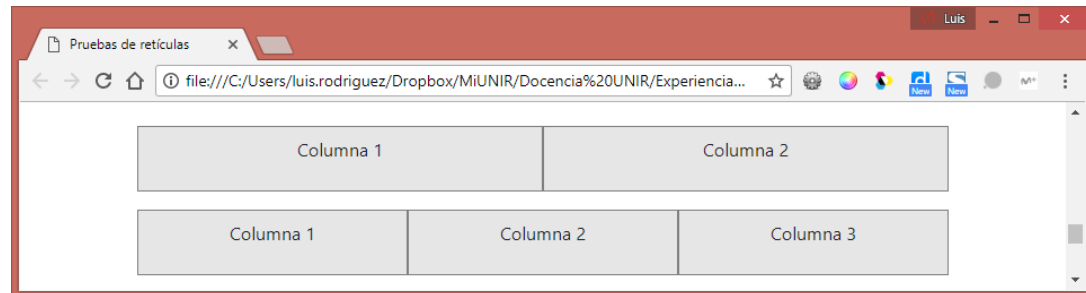
	Dispositivos muy pequeños (576px)	Dispositivos muy pequeños Teléfonos (<768px)	Dispositivos pequeños Tabletas (≥768px)	Dispositivos de tamaño mediano Computadoras de escritorios (≥992px)	Dispositivos grandes Computadoras de escritorio (≥1200px)
Ancho máximo del contenedor	Ninguno (automático)	540px	720px	960px	1140px
Prefijo de la clase	.col	.col-sm	.col-md	.col-lg	.col-xl
núm. de columnas	12				
Ancho de separación	30px (15px en cada lado de una columna)				
Permite anidado	Sí				
Desplazamiento	Sí				
Cambio de orden de columnas	Sí				

- Todos pueden llevar el sufijo – *, para especificar el número de columnas de la retícula que ocupará el elemento.

Ancho automático de columnas

- ▶ Si no se incluye el número de columnas que ocupará el bloque, el ancho se calcula automáticamente.
- ▶ El ancho se puede repartir de forma equitativa si todas las columnas pertenecen a la clase `.col` o `.col-*` (* hace referencia al tipo de dispositivo sm, md, lg o xl).

```
<div class="row">
  <div class="col">
    <p>Columna 1</p>
  </div>
  <div class="col">
    <p>Columna 2</p>
  </div>
</div>
<div class="row">
  <div class="col">
    <p>Columna 1</p>
  </div>
  <div class="col">
    <p>Columna 2</p>
  </div>
  <div class="col">
    <p>Columna 3</p>
  </div>
</div>
```

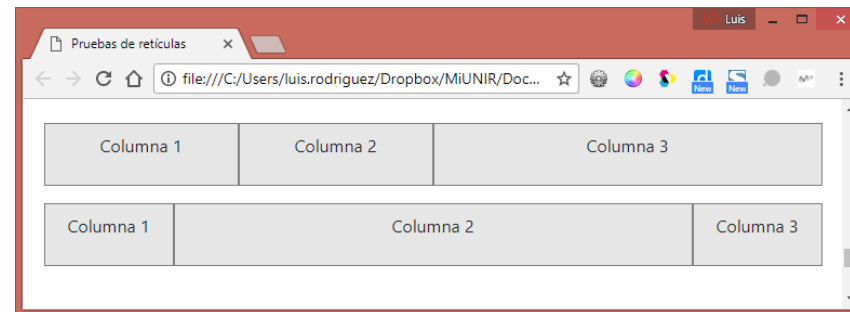
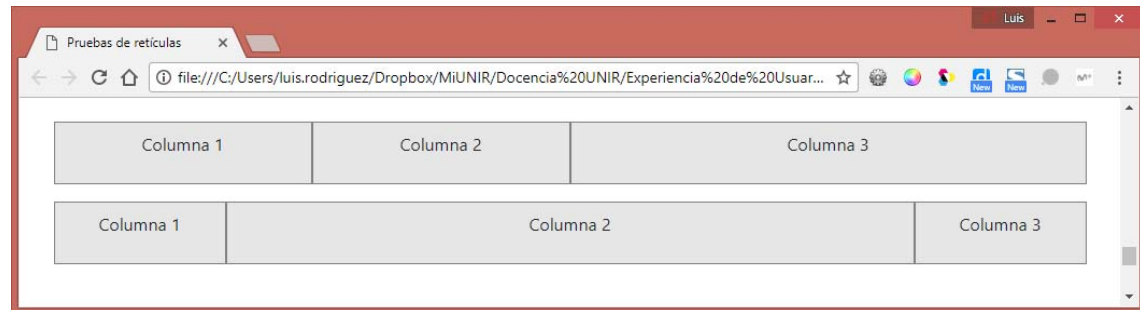


La primera fila reparte el ancho entre las dos columnas que contiene
La segunda fila reparte el ancho disponible entre las tres columnas que contiene

Ancho automático de columnas (II)

- Si se especifica el ancho solo algunas columnas, el resto se ajusta de forma automática.

```
<div class="row">
  <div class="col">
    <p>Columna 1</p>
  </div>
  <div class="col">
    <p>Columna 2</p>
  </div>
  <div class="col-6">
    <p>Columna 3</p>
  </div>
</div>
<div class="row">
  <div class="col">
    <p>Columna 1</p>
  </div>
  <div class="col-4">
    <p>Columna 2</p>
  </div>
  <div class="col">
    <p>Columna 3</p>
  </div>
</div>
```



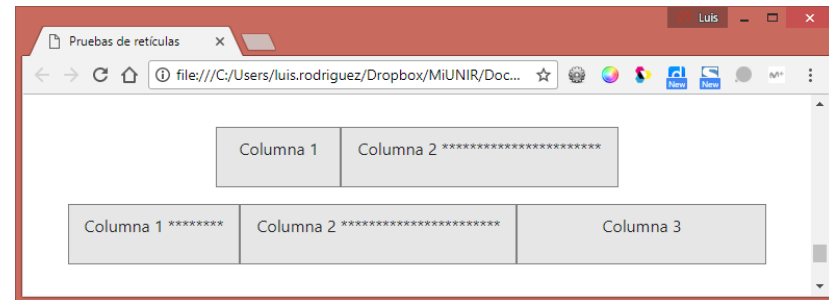
En la primera fila la columna 3 tiene ocupa 6 columnas. Las columnas 1 y 2 reparten el espacio de forma equitativa.

En la segunda fila la columna 2 ocupa 8 columnas. La columnas 1 y 3 reparten el espacio de forma equitativa

Ancho automático de columnas (III)

- El sufijo `-auto` en las clases de las columnas, ajusta el ancho de forma automática al contenido.

```
<div class="row justify-content-center">
<!-- justify-content-center ajusta el contenido
    al centro del contenedor
-->
  <div class="col-2">
    <p>Columna 1</p>
  </div>
  <div class="col-auto">
    <p>Columna 2 *****</p>
  </div>
</div>
<div class="row justify-content-center">
  <div class="col-auto">
    <p>Columna 1 *****</p>
  </div>
  <div class="col-auto">
    <p>Columna 2 *****</p>
  </div>
  <div class="col-4">
    <p>Columna 3</p>
  </div>
</div>
```



En la primera fila, la segunda columna de ajusta el ancho al contenido de la misma.

En la segunda fila, las columnas 1 y 2 ajustan su ancho al contenido.

Diseño responsive

- ▶ Los cinco niveles predefinidos de clases permiten ajustar el diseño al ancho del *viewpoint* de los distintos dispositivos.
- ▶ Para indicar que un elemento ocupe un número de columnas distinto dependiendo del tipo de dispositivo hay que utilizar las clases con el especificador `sm`, `md`, `lg` o `xl`, o sin ninguno de ellos para los dispositivos muy pequeños.
- ▶ Por ejemplo, si se desea que un elemento `div` ocupe 12 columnas para dispositivos muy pequeños, 6 para dispositivos pequeños y 4 para grandes, habría que codificarlo:

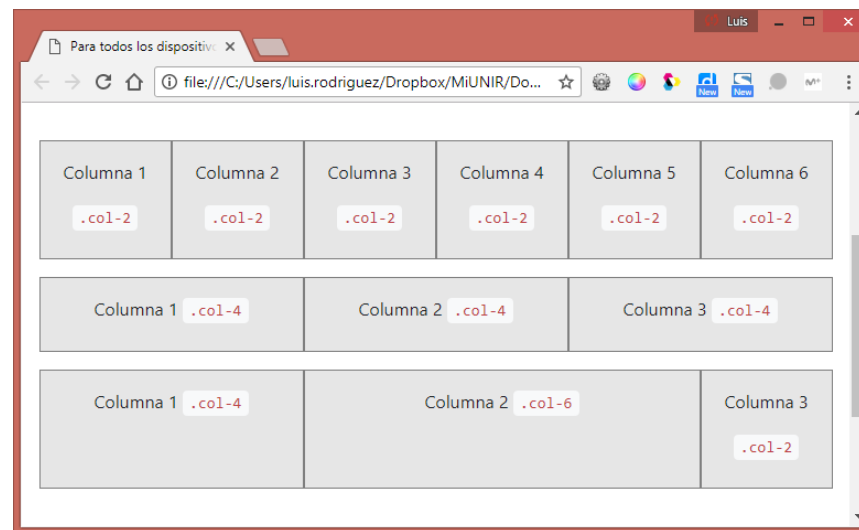
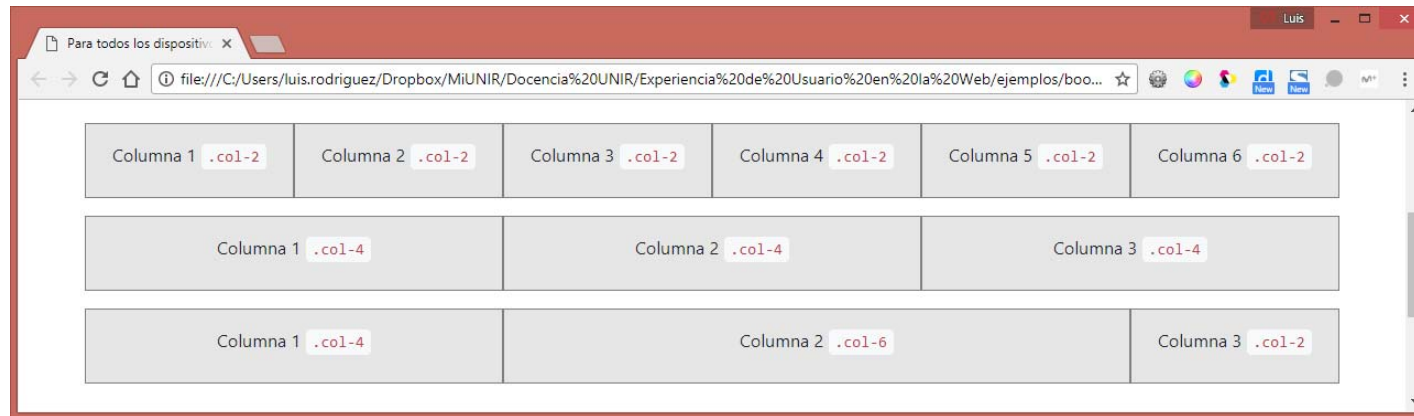
```
<div class="col-12 col-sm-6 col-lg-4">
```

- La clase sin especificador hace referencia a los dispositivos muy pequeños y hará que en esos dispositivos el elemento ocupe 12 columnas de la retícula.
- La clase con el especificador `sm` hace referencia a los dispositivos pequeños y hará que en esos dispositivos de más de 575px el elemento ocupe 6 columnas.
- La clase con el especificador `lg` hace referencia a los dispositivos grandes y hará que en los dispositivos de 992px y más el elemento ocupe 4 columnas.
- Los puntos de ruptura se basan en anchos mínimos por lo que:
 - En los dispositivos muy pequeños (< 576px) el elemento ocuparía 12 columnas.
 - En los pequeños y medianos (>=576px y <992px) el elemento ocuparía 6 columnas.
 - En los grandes y muy grandes (>=992px) el elemento ocuparía 4 columnas.

Ejemplo: igual para todos los dispositivos

- ▶ Para que el *layout* de la página sea fijo en todos los dispositivos se utilizan las clases `.col` o `.col-*`.
- ▶ Las columnas adaptarían el ancho al ancho del contenedor pero no cambiaría su estructura según el tamaño del *viewport*.
- ▶ En el ejemplo, independientemente del dispositivo, siempre:
 - La primera fila tiene 6 bloques que ocupan dos columnas de la retícula cada uno.
 - La segunda fila tiene 3 bloques que ocupan cuatro columnas de la retícula cada uno.
 - La tercera fila tiene 3 bloques que ocupan, respectivamente, 4, 6 y 2 bloques de la retícula.

Ejemplo: igual para todos los dispositivos (II)



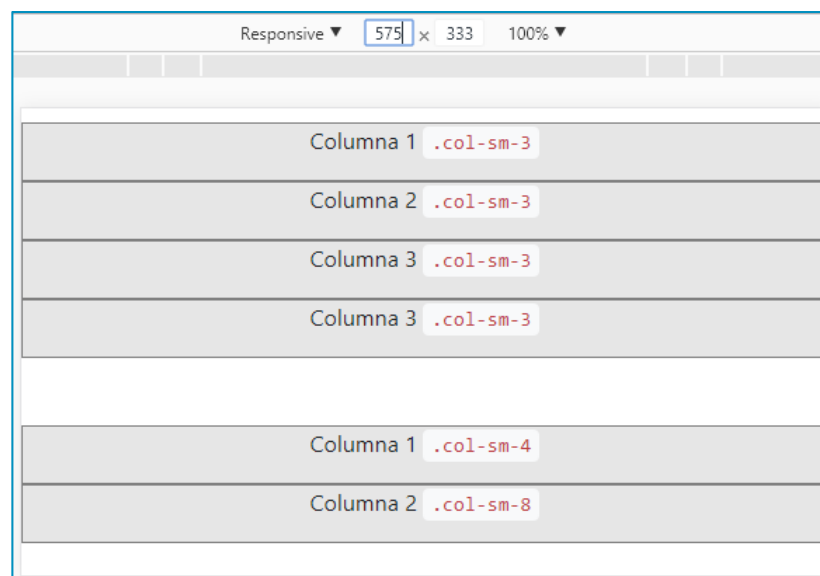
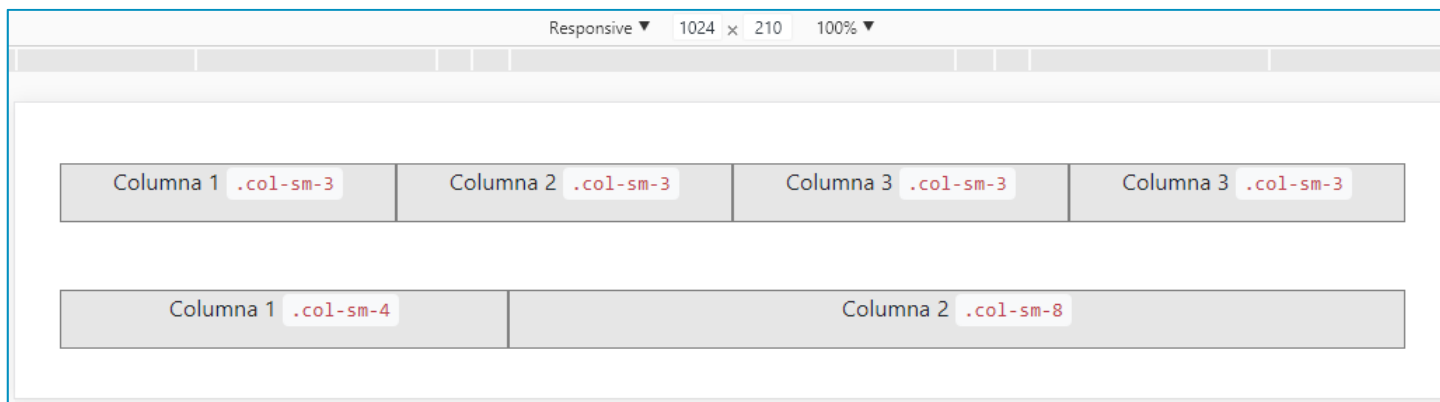
Ejemplo: igual para todos los dispositivos (III)

```
<div class="row">
  <!-- El siguiente bloque se repite 6 veces -->
  <div class="col-2">
    <p>Columna 1 <code>.col-2</code></p>
  </div>
  ...
<div class="row">
  <!-- El siguiente bloque se repite 3 veces -->
  <div class="col-4">
    <p>Columna 1 <code>.col-4</code></p>
  </div>
  ...
</div>
<div class="row">
  <div class="col-4">
    <p>Columna 1 <code>.col-4</code></p>
  </div>
  <div class="col-6">
    <p>Columna 2 <code>.col-6</code></p>
  </div>
  <div class="col-2">
    <p>Columna 3 <code>.col-2</code></p>
  </div>
</div>
```

Ejemplo: apilar el contenido en dispositivos muy pequeños

- ▶ A la hora de diseñar una estructura de página *responsive*, es habitual que en los dispositivos más pequeños los distintos bloques de contenidos aparezcan unos encima de otros.
- ▶ Cuando se incluye alguna de las clases de columna con indicador -sm, -md, -lg o -xl, si el ancho del *viewport* es menor que el punto de ruptura los bloques se apilarán unos encima de otros.
 - Si se utilizan las clases col-*-sm, se podrá crear un diseño especial que distinga los dispositivos muy pequeños del resto.
- ▶ En el ejemplo,
 - Si el *viewport* es mayor de 575px.
 - La primera fila está dividida en cuatro bloques que ocupan 3 columnas cada uno.
 - La segunda fila está dividida en dos bloques que ocupan 4 y 8 columnas respectivamente.
 - Si el *viewport* es menor o igual de 575px.
 - Los bloques de las dos filas se apilan unos encima de otros.

Ejemplo: apilar el contenido en dispositivos muy pequeños (II)



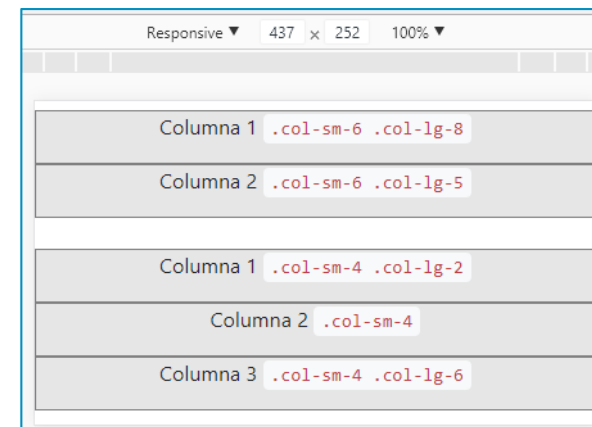
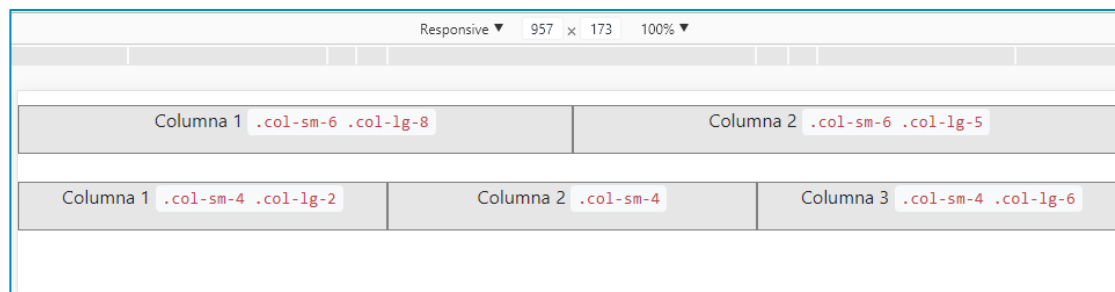
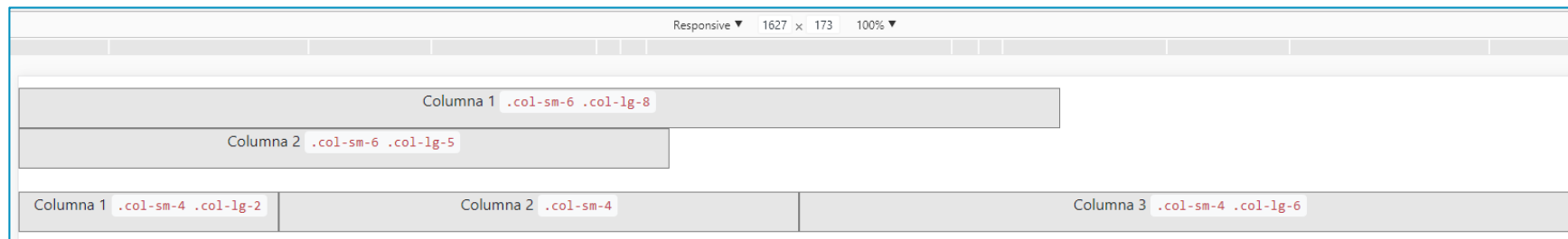
Ejemplo: apilar el contenido en dispositivos muy pequeños (III)

```
<div class="row">
  <div class="col-sm-3">
    <p>Columna 1 <code>.col-sm-3</code></p>
  </div>
  <div class="col-sm-3">
    <p>Columna 2 <code>.col-sm-3</code></p>
  </div>
  <div class="col-sm-3">
    <p>Columna 3 <code>.col-sm-3</code></p>
  </div>
  <div class="col-sm-3">
    <p>Columna 3 <code>.col-sm-3</code></p>
  </div>
</div>
<div class="row">
  <div class="col-sm-4">
    <p>Columna 1 <code>.col-sm-4</code></p>
  </div>
  <div class="col-sm-8">
    <p>Columna 2 <code>.col-sm-8</code></p>
  </div>
</div>
```

Ejemplo: diseño para varios tipos de dispositivos

- ▶ Para conseguir una estructura de página distinta para distintos dispositivos (por ejemplo, para móviles pequeños, tabletas y dispositivos de escritorio), se pueden mezclar varias clases `.col*` en un mismo elemento.
- ▶ En el ejemplo:
 - En la primera fila:
 - En dispositivos muy pequeños el contenido se distribuye en bloques apilados.
 - En dispositivos pequeños y medianos el contenido se distribuye en 2 bloques iguales que ocupan 6 columnas de la retícula.
 - En dispositivos grandes y muy grandes el contenido se distribuye en dos bloques que ocupan 8 y 5 columnas de la retícula.
 - Dado que la suma de columnas es mayor que 12, se produce un salto de línea en el segundo bloque.
 - Si la suma de las columnas es menor que 12, las columnas no llegan a ocupar todo el espacio del contenedor.
 - En la segunda fila:
 - En dispositivos muy pequeños el contenido se distribuye en bloques apilados.
 - En dispositivos pequeños y medianos el contenido se distribuye en 3 bloques iguales.
 - En dispositivos grandes y muy grandes el contenido se distribuye en 3 bloques que ocupan 2, 4 y 6 columnas de la retícula.

Ejemplo: diseño para varios tipos de dispositivos (II)



Ejemplo: diseño para varios tipos de dispositivos (III)

```
<div class="row">
  <div class="col-sm-6 col-lg-8">
    <p>Columna 1 <code>.col-sm-6 .col-lg-8</code></p>
  </div>
  <div class="col-sm-6 col-lg-5">
    <p>Columna 2 <code>.col-sm-6 .col-lg-5</code></p>
  </div>
</div>
</br>
<div class="row">
  <div class="col-sm-4 col-lg-2">
    <p>Columna 1 <code>.col-sm-4 .col-lg-2</code></p>
  </div>
  <div class="col-sm-4">
    <p>Columna 2 <code>.col-sm-4</code></p>
  </div>
  <div class="col-sm-4 col-lg-6">
    <p>Columna 3 <code>.col-sm-4 .col-lg-6</code></p>
  </div>
</div>
```

Alineación de los bloques

- Para la alineación vertical de los bloques dentro de un contenedor `.row`, se pueden utilizar las clases `.align-items-start`, `.align-items-center` y `.align-items-end`.

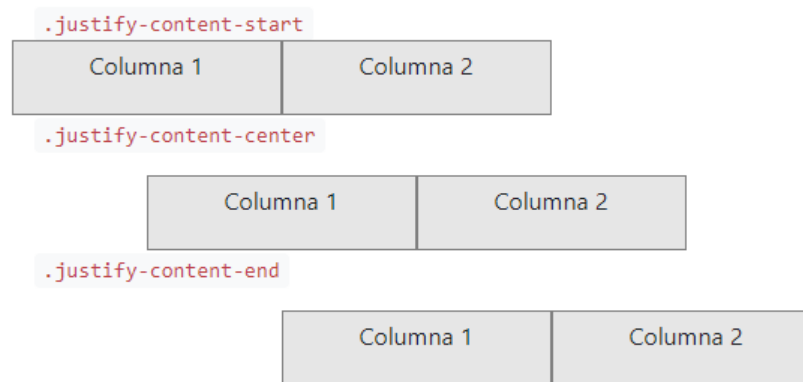
```
<div class="container">
  <div class="row align-items-start">
    <div class="col">
      <p>.row contiene <code>.align-items-start</code></p>
    </div>
  </div>
</div>
<div class="container">
  <div class="row align-items-center">
    <div class="col">
      <p>.row contiene <code>.align-items-center</code></p>
    </div>
  </div>
</div>
<div class="containeralto">
  <div class="row align-items-end">
    <div class="col">
      <p>.row contiene <code>.align-items-end</code></p>
    </div>
  </div>
</div>
```



Alineación de los bloques (II)

- La alineación horizontal se hace con las clases `.justify-content-start`, `.justify-content-center` y `.justify-content-end` en un elemento de la clase `.row`.

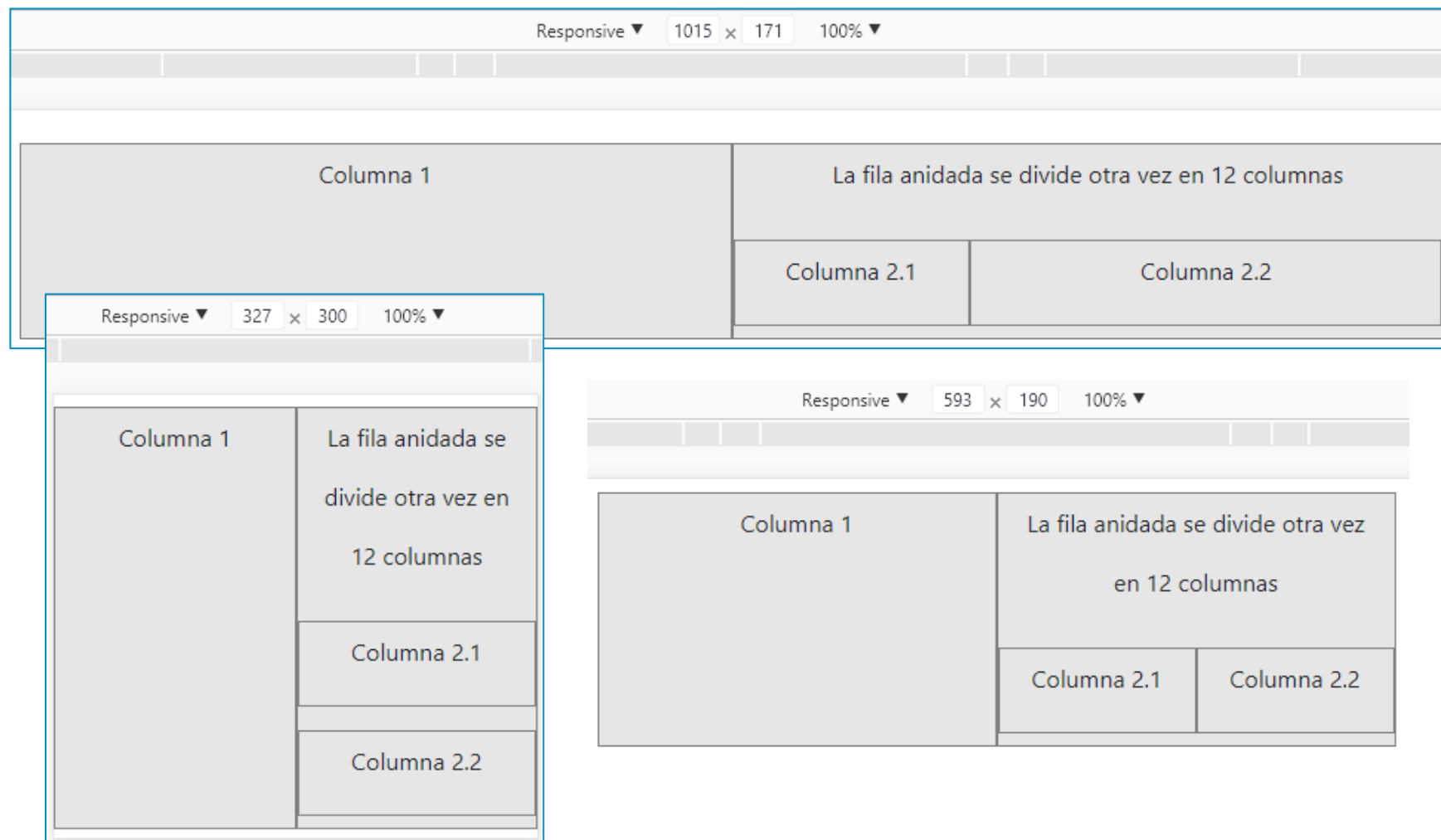
```
<div class="row justify-content-start">
  <div class="col-4">
    <p>Columna 1</p>
  </div>
  <div class="col-4">
    <p>Columna 2</p>
  </div>
</div>
<div class="row justify-content-center">
  <div class="col-4">
    <p>Columna 1</p>
  </div>
  <div class="col-4">
    <p>Columna 2</p>
  </div>
</div>
<div class="row justify-content-end">
  <div class="col-4">
    <p>Columna 1</p>
  </div>
  <div class="col-4">
    <p>Columna 2</p>
  </div>
</div>
```



Anidamiento de columnas

- ▶ Se pueden anidar columnas, una dentro de otra.
- ▶ Para anidar columnas hay que definir un elemento de la clase *row* dentro de una columna.
 - Cuando se crea un elemento de la clase *row* dentro de una columna, esta se vuelve a dividir en 12 columnas.
- ▶ En el siguiente ejemplo:
 - Las columnas «padre» tienen siempre el mismo ancho.
 - En dispositivos muy pequeños, los bloques de las columnas anidadas se apilan.
 - En dispositivos pequeños y medianos, los bloques de las columnas anidadas ocupan seis columnas de la retícula cada uno.
 - En dispositivos grandes y muy grandes, los bloques de las columnas anidadas ocupan 4 y 8 columnas de la retícula cada uno.

Anidamiento de columnas (II)



Anidamiento de columnas (III)

```
<div class="row">
  <div class="col-6">
    <p>Columna 1</p>
  </div>
  <div class="col-6">
    <p>La fila anidada se divide otra vez en 12 columnas</p>
    <div class="row">
      <div class="col-sm-6 col-md-4">
        <p>Columna 2.1</p>
      </div>
      <div class="col-sm-6 col-md-8">
        <p>Columna 2.2</p>
      </div>
    </div>
  </div>
</div>
```

Reordenar columnas

- ▶ Al mostrar la página, las columnas aparecen en el orden del código.
- ▶ Las clases `.order-n` y `.order-*-n` reordenan las columnas.
 - `*` puede tomar los valores `sm`, `md`, `lg` o `xl`.
 - `n` representa el orden del bloque dentro del contenedor. Puede tomar valores entre 1 y 12.

```
<div class="row">
  <div class="col order-12">
    <p>Columna 1 <code>.order-12</code></p>
  </div>
  <div class="col order-4">
    <p>Columna 2 <code>.order-4</code></p>
  </div>
  <div class="col order-8">
    <p>Columna 3 <code>.order-8</code></p>
  </div>
</div>
```

Columna 2	Columna 3	Columna 1
<code>.order-4</code>	<code>.order-8</code>	<code>.order-12</code>

Un ejemplo simple

- ▶ Se trata de la típica estructura que consta de:
 - Una cabecera.
 - Un menú de navegación lateral.
 - Un área central con el contenido principal.
 - Un área de contenido relacionado.
 - Un pie de página.
- ▶ Además de los elementos que se han visto el ejemplo utiliza:
 - Elementos de navegación:
 - La clase `.nav .flex-column` permite hacer un menú de navegación vertical.
 - Cada elemento de menú utiliza la clase `.nav-link`.
 - Alineación de texto:
 - Las clases `.text-left`, `.text-center` y `.text-right` permiten alinear el texto.
 - Las clases `.text-*-left`, `.text-*-center` y `.text-*-right`, permiten aplicar alineación izquierda, central o derecha dependiendo de los puntos de ruptura de las media queries.
 - * puede tomar los valores `sm`, `md`, `lg` y `xl`.

Un ejemplo simple (II)

Ejemplo simple

[Enlace 1](#)
[Enlace 2](#)
[Enlace 3](#)
[Enlace 4](#)

Contenidos

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim.

Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, sem quam semper libero, sit amet adipiscing sem neque sed ipsum. Nam quam nunc, blandit vel, luctus pulvinar, hendrerit id, lorem. Maecenas nec odio et ante tincidunt tempus. Donec vitae sapien ut libero venenatis faucibus. Nullam quis ante. Etiam sit amet orci eget eros faucibus tincidunt. Duis leo. Sed fringilla mauris sit amet nibh. Donec sodales sagittis magna. Sed consequat, leo eget bibendum sodales, augue velit cursus nunc, quis gravida magna mi a libero. Fusce vulputate eleifend sapien. Vestibulum purus quam, scelerisque ut, mollis sed, nonummy id, metus. Nullam accumsan lorem in dui. Cras ultricies mi eu turpis hendrerit fringilla. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; In ac dui quis mi consectetur lacinia.

Nam pretium turpis et arcu. Duis arcu tortor, suscipit eget, imperdiet nec, imperdiet iaculis, ipsum. Sed aliquam ultrices mauris. Integer ante arcu, accumsan a, consectetur eget, posuere ut, mauris. Praesent adipiscing. Phasellus ullamcorper ipsum rutrum nunc. Nunc nonummy metus. Vestibulum volutpat pretium libero. Cras id dui. Aenean ut eros et nisl sagittis vestibulum. Nullam nulla eros, ultricies sit amet, nonummy id, imperdiet feugiat, pede. Sed lectus. Donec mollis hendrerit risus. Phasellus nec sem in justo pellentesque facilisis. Etiam imperdiet imperdiet orci. Nunc nec neque. Phasellus leo dolor, tempus non, auctor et, hendrerit quis, nisi.

Curabitur ligula sapien, tincidunt non, euismod vitae, posuere imperdiet, leo. Maecenas malesuada. Praesent congue erat at massa. Sed cursus turpis vitae tortor, Donec posuere vulputate arcu. Phasellus accumsan cursus velit. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Sed aliquam, nisl quis porttitor congue, elit erat euismod orci, ac placerat dolor lectus quis orci. Phasellus consectetur vestibulum elit, Aenean tellus metus, bibendum sed, posuere ac, mattis non, nunc. Vestibulum fringilla pede sit amet augue. In turpis. Pellentesque posuere. Praesent turpis.

Aenean posuere, tortor sed cursus feugiat, nunc augue blandit nunc, eu sollicitudin urna dolor sagittis lacus. Donec elit libero, sodales nec, volutpat a, suscipit non, turpis. Nullam sagittis. Suspendisse pulvinar, augue ac venenatis condimentum, sem libero volutpat nibh, nec pellentesque velit pede quis nunc. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Fusce id purus. Ut varius tincidunt libero. Phasellus dolor. Maecenas vestibulum mollis diam. Pellentesque ut neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

In dui magna, posuere eget, vestibulum et, tempor auctor, justo. In ac felis quis tortor malesuada pretium. Pellentesque auctor neque nec urna. Proin sapien ipsum, porta a, auctor quis, euismod ut, mi. Aenean viverra rhoncus pede. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Ut non enim eleifend felis pretium feugiat. Vivamus quis mi. Phasellus a est. Phasellus magna.

Otras cosas

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim.

© Una empresa cualquiera, 2018

En dispositivos grandes y muy grandes

Ejemplo simple

[Enlace 1](#)
[Enlace 2](#)
[Enlace 3](#)
[Enlace 4](#)

Contenidos

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim.

Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, sem quam semper libero, sit amet adipiscing sem neque sed ipsum. Nam quam nunc, blandit vel, luctus pulvinar, hendrerit id, lorem. Maecenas nec odio et ante tincidunt tempus. Donec vitae sapien ut libero venenatis faucibus. Nullam quis ante. Etiam sit amet orci eget eros faucibus tincidunt. Duis leo. Sed fringilla mauris sit amet nibh.

Viverra rhoncus pede. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Ut non enim eleifend felis pretium feugiat. Vivamus quis mi. Phasellus a est. Phasellus magna.

Otras cosas

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim.

© Una empresa cualquiera, 2018

En dispositivos medianos

Ejemplo simple

[Enlace 1](#)
[Enlace 2](#)
[Enlace 3](#)
[Enlace 4](#)

Contenidos

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim.

Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, sem quam semper libero, sit amet adipiscing sem neque sed ipsum. Nam quam nunc, blandit vel, luctus pulvinar, hendrerit id, lorem. Maecenas nec odio et ante tincidunt tempus. Donec vitae sapien ut libero venenatis faucibus. Nullam quis ante. Etiam sit amet orci eget eros faucibus tincidunt. Duis leo. Sed fringilla mauris sit amet nibh.

Viverra rhoncus pede. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Ut non enim eleifend felis pretium feugiat. Vivamus quis mi. Phasellus a est. Phasellus magna.

Otras cosas

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim.

© Una empresa cualquiera, 2018

En dispositivos pequeños

Un ejemplo simple (III)

```
<!doctype html>
<html lang="es">
<head>
  <title>Ejemplo simple</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- Llamada al nucleo de bootstrap-->
  <link rel="stylesheet" href="css/bootstrap.min.css">
  <!-- Aquí pueden ir las hojas de estilo propias -->
  <style>
    header{
      margin-bottom:1rem;
      padding-bottom:0.5rem;
      border-bottom: black thin solid;
    }
    footer{
      margin-top:1rem;
      padding-top:0.5rem;
      border-top: black thin solid;
      text-align:center;
    }
  </style>
</head>
<body>
  <div class="container">
    <header class="row">
      <h1 class="col text-center text-sm-left">Ejemplo simple</h1>
    </header>
```

Un ejemplo simple (IV)

```
<div class="row">
  <nav class="col-sm-3 col-md-2">
    <ul class="nav flex-column text-center text-sm-left">
      <li><a href="#" class="nav-link">Enlace 1</a></li>
      <li><a href="#" class="nav-link">Enlace 2</a></li>
      <li><a href="#" class="nav-link">Enlace 3</a></li>
      <li><a href="#" class="nav-link">Enlace 4</a></li>
    </ul>
  </nav>
  <main class="col-sm-9 col-lg-10">
    <div class="row">
      <article class="col-lg-8">
        <h2 class="text-center text-sm-left">Contenidos</h2>
        <p>Lorem ipsum dolor sit amet, [...] Phasellus a est. Phasellus magna.</p>
      </article>
      <aside class="col-lg-4">
        <h2 class="text-center text-sm-left">Otras cosas</h2>
        <p>Lorem ipsum dolor sit amet, [...], consequat vitae, eleifend ac, enim. </p>
      </aside>
    </div>
  </main>
</div>
```

Un ejemplo simple (V)

```
<footer class="row">
  <p class="col">&copy; Una empresa cualquiera, 2018</p>
</footer>
</div>
<!-- Aquí se colocan los scripts para que la página se cargue más rápido -->
<script src="js/jquery-3.2.1.min.js"></script>
<script src="js/popper.js"></script>
<script src="js/bootstrap.min.js"></script>
</body>
</html>
```

- En <https://goo.gl/hmBg4X> se puede encontrar el ejemplo completo.



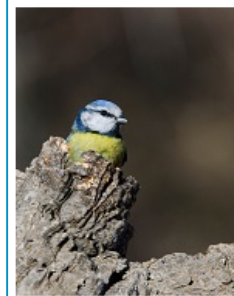
Imágenes y flotantes

Imágenes

- ▶ Por omisión, las imágenes no son *responsive*.
 - Normalmente este es el funcionamiento correcto cuando se trata de incluir imágenes flotantes dentro de un párrafo.
 - Para hacer que floten y se ajusten dentro de un párrafo para ajustarlas dentro del párrafo se utilizan las clases flotantes: `.float-left`, `.float-right` y `.float-none`.
 - Son las equivalentes a los atributos CSS `float:left`, `float:right` y `float:none`.
 - La alineación también se puede aplicar en el diseño *responsive* con las clases `.float-*-left`, `.float-*-right` y `.float-*-none`.
 - Como siempre, `*` puede tomar los valores `sm`, `md`, `lg` o `xl`.
 - Las clases flotantes se pueden aplicar a todos los elementos HTML.

Imágenes (II)

```
<p>Lorem ipsum dolor sit amet,...</p>
<!-- la clase pr-2 dejar un padding a la derecha de la imagen
-->
```



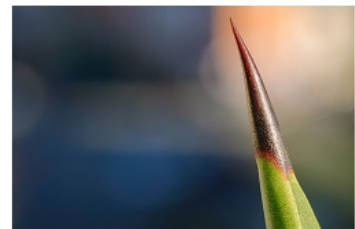
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque euismod. Quis enim.

Etiam rhoncus. Maecenas tempus, tincidunt ante orci a arcu. Mauris nibh. Maecenas nec odio et magna tincidunt magna. Nullam quis ante. Etiam

```
<p>Etiam sit amet eget ... amet
nibh.</p>
<!-- la clase pl-2 dejar un padding a la derecha de la imagen
-->
```

ligula eget dolor. Maecenas diam, diam. Mauris nibh. Maecenas nec odio et magna tincidunt magna. Nullam quis ante. Etiam

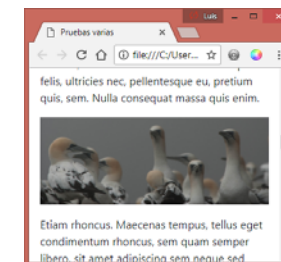
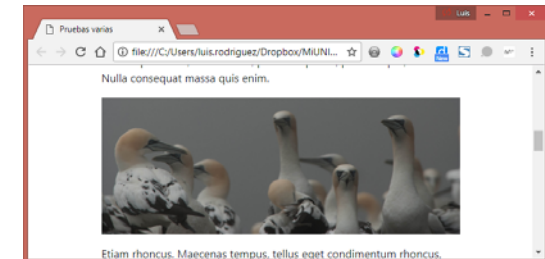
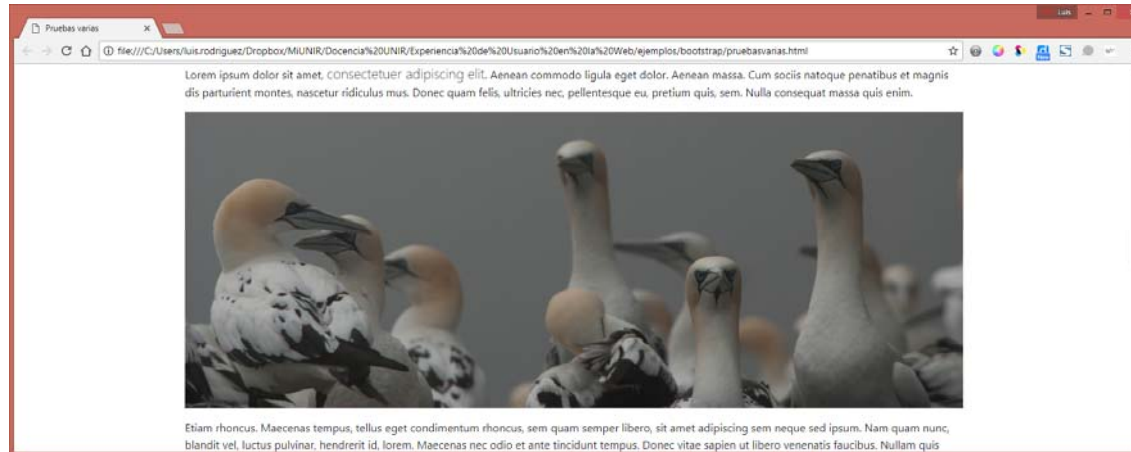
Etiam rhoncus. Maecenas tempus, tincidunt ante orci a arcu. Mauris nibh. Maecenas nec odio et magna tincidunt magna. Nullam quis ante. Etiam



Imágenes (III)

- Se pueden conseguir imágenes *responsive* con la clase `.img-fluid` que permite adaptar la imagen al ancho del contenedor.

```
... pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. </p>  
<p></p>  
<p>Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, ...
```



Imágenes (IV)

- ▶ La clase `.img-thumbnail` enmarca la imagen, al tiempo que también permite que se adapte su tamaño al ancho del contenedor, creando también imágenes *responsive*.
 - El borde del marco es redondeado, pero se puede cambiar con las clases relacionadas con la propiedad `border-radius` de CSS.
 - `.rounded`, redondea las esquinas del elemento.
 - `.rounded-top`, `-right`, `-bottom` y `-left`, redondea las esquinas superiores, derechas, inferiores e izquierdas.
 - `.rounded-circle`, enmarca el elemento en un círculo.
 - `.rounded-0`, elimina el efecto de redondeo.



`class="img-thumbnail"`



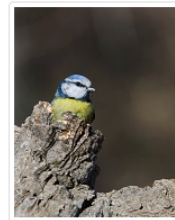
`class="rounded-circle"`

Restablecer flotantes

- ▶ Cuando se utilizan elementos flotantes, los bloques se pueden montar unos sobre otros.
 - Por ejemplo, si la imagen corresponde a la «Sección 1», sería interesante que el título «Sección 2» apareciera en una nueva línea.
 - El atributo `clear` de `css` realiza esa función, restableciendo los flotantes.
- ▶ En Bootstrap, la clase `.clearfix` realiza esa función.

```
<p> Lorem ipsum dolor...
ridiculus mus.</p>
<p>Etiam rhoncus...luctus pulvinar.</p>
<div class="clearfix"></div>
<h2>Sección 2</h2>
<p>Etiam rhoncus. Maecenas tempus, ...
```

Sección 1



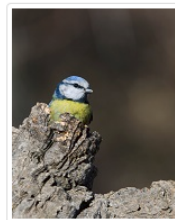
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean con
penatibus et magnis dis parturient montes, nascetur ridiculus mus.

Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, si
ipsum. Nam quam nunc, blandit vel, luctus pulvinar.

Sección 2

Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, si
ipsum. Nam quam nunc, blandit vel, luctus pulvinar, hendrerit id, lorem
sapien ut libero venenatis faucibus. Nullam quis ante. Etiam sit amet orci eget eros faucibus tin

Sección 1



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean comm
penatibus et magnis dis parturient montes, nascetur ridiculus mus.

Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, se
ipsum. Nam quam nunc, blandit vel, luctus pulvinar.

Sección 2

Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, sem quam semper libero, s
blandit vel, luctus pulvinar, hendrerit id, lorem. Maecenas nec odio et ante tincidunt tempus. Doi

Ocultar y mostrar contenido

Ocultar y mostrar contenido

- ▶ En ocasiones puede ser conveniente que ocultar o mostrar contenido para determinados dispositivos.
- ▶ Las clases `.d-*` permiten cambiar los valores de la propiedad `display` de `css`.
 - Las más utilizadas:
 - `.d-none`, equivale a `display:none`. Elimina el elemento del árbol DOM, por lo que no mostrará su contenido ni será accesible a ningún agente de usuario.
 - `.d-block`, equivale a `display:block`. Convierte el elemento en un elemento de bloque. Si el elemento está invisible lo muestra.
 - `.d-inline`, equivale a `display:inline`. Convierte el elemento en un elemento en línea. Si el elemento está invisible lo muestra.
 - Todas admiten los modificadores responsive `-sm-`, `-md-`, `-lg-` y `-xl-`.

Ocultar y mostrar contenido (II)

- ▶ Un menú con un número elevado de elementos en la cabecera puede ser adecuado para pantallas grandes, pero no en pantallas pequeñas.
 - Puede ser interesante, eliminar las opciones menos importantes o a las que se puede acceder de otra forma en los dispositivos pequeños.
 - Utilizando la clase `.d-none` el elemento no se visualizaría. Combinado con la clase `.d-sm-block` o `.d-sm-inline` (según se trate de un elemento de bloque o en línea) esas opciones se eliminarían para los dispositivos muy pequeños , quedando visibles para el resto.
- ▶ La clase `.sr-only` se utiliza para el elemento no sea visible pero que esté disponible los agentes de usuario.
 - Saca el bloque de área de visualización, aunque se mantiene en el árbol DOM.
 - El elemento estaría disponible, por ejemplo, para los lectores de pantalla.
 - Esto puede ser útil si, por cuestiones de diseño, no queremos que se vea un elemento, pero sí queremos que esté disponible para los agentes de usuario no visuales.

Ejemplo



- ▶ El ejemplo se puede ver en <https://goo.gl/kP6J3T>
- ▶ Utiliza las clases *responsive* de Bootstrap para adaptar la página a los distintos dispositivos.
 - Con pantallas grandes y muy grandes:
 - Presenta una estructura de tres columnas.
 - Las noticias se agrupan en dos filas de tres bloques cada una.
 - Con pantallas pequeñas y medias:
 - Las noticias se agrupan en tres filas de dos noticias cada una.
 - La columna de la derecha aparece después de las noticias.
 - Con pantallas muy pequeñas:
 - Las distintas secciones se apilan (artículo, noticias, relacionado).
 - La imagen del artículo principal se centra en el texto.
 - La barra de navegación “colapsa” y las opciones se despliegan al pulsar el botón izquierdo de tipo «hamburger».

66

Ejemplo (III)

```
<!doctype html>
<html lang="es">
<head>
  <title>Ejemplo con imágenes</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- Llamada al nucleo de bootstrap-->
  <link rel="stylesheet" href="css/bootstrap.min.css">
  <!-- Aquí pueden ir las hojas de estilo de usuario -->
</head>
<body>
  <div class="container">
    <header class="row pb-4">
      <div class="col">
        <h1 class="text-center text-sm-left">Ejemplo <small class="muted">con imágenes</small></h1>
        <!-- La cabecera no se ve, pero está disponible para los lectores de pantalla -->
        <h2 class="sr-only">Menú de navegación principal</h2>
        <!-- Barra de navegación, se expande a partir de pantallas pequeñas.
              navbar-dark bg-primary es para los esquemas de color -->
        <nav class="navbar navbar-expand-sm navbar-dark bg-primary">
          <!-- Aparece un botón de tipo "hamburger" en pantallas pequeñas con atributos WAI-ARIA -->
          <button class="navbar-toggler" type="button" data-toggle="collapse"
            data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
            aria-expanded="false" aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
          </button>
```

Ejemplo (IV)

```
<!-- El contenido de los elementos de menú expandido y en el botón hamburguer -->
<div class="collapse navbar-collapse" id="navbarSupportedContent">
  <ul class="navbar-nav mr-auto">
    <li class="nav-item"><a class="nav-link" href="#">Inicio</a></li>
    <li class="nav-item"><a class="nav-link active" href="#">Artículo</a></li>
    <li class="nav-item"><a class="nav-link" href="#">Documentos</a></li>
    <li class="nav-item"><a class="nav-link" href="#">Productos</a></li>
    <li class="nav-item"><a class="nav-link" href="#">Acerca de</a></li>
  </ul>
</div>
</nav>
</div>
</header>
<div class="row">
  <main class="col-12 col-lg-9">
    <div class="row">
      <article class="col">
        <h2 class="text-center text-sm-left">Artículo principal</h2>
        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit... massa quis enim. </p>
        <p>Donec pede justo, fringilla vel, aliquet nec, vulputate ... eleifend ac, enim. </p>
        <p>
          Etiam rhoncus... consectetur lacinia.</p>
      </article>
    </div>
  </main>
</div>
```

Ejemplo (V)

```
<section class="row">
  <article class="col-12 col-sm-6 col-lg-4 pb-3">
    <h3 class="text-center text-sm-left">Noticia 1</h3>
    
    <p>Lorem ipsum dolor sit amet, ... quis enim. </p>
    <p><a class="btn btn-light btn-outline-dark float-right" href="#" role="button">
      Ver m&aacute;s...</a></p>
  </article>
<div class="clearfix"></div>
<!-- Esta estructura se repita para las 6 noticias -->
<article>
  <h3 class="text-center text-sm-left">Noticia 6</h3>
  
  <p>Lorem ipsum dolor sit amet, ... quis enim. </p>
  <p><a class="btn btn-light btn-outline-dark float-right" href="#" role="button">
    Ver m&aacute;s...</a></p>
</article>
</section>
</main>
<aside class="col-12 col-lg-3">
  <h2 class="text-center text-sm-left">Relacionado</h2>
  <p>Lorem ipsum dolor sit amet, ... consequat massa quis enim. </p>
  <p>Donec pede justo, ... eleifend ac, enim. </p>
</aside>
</div>
```

Ejemplo (VI)

```
<footer class="row">
  <p class="col-12 text-center text-white bg-primary mt-4">
    &copy; Una empresa cualquiera, 2018
  </p>
</footer>
</div>
<!-- Aquí se colocan los scripts para que la página se cargue más rápido -->
<script src="js/jquery-3.2.1.min.js"></script>
<script src="js/popper.js"></script>
<script src="js/bootstrap.min.js"></script>
</body>
</html>
```

Ejemplo (VIII)

► Nuevas clases utilizadas.

- Alineación (getbootstrap.com/docs/4.0/utilities/text/#text-alignment)
 - .text-center: centra el contenido
 - .text-sm-left: en pantallas con un ancho del viewport superior a 575px, el contenido se alinea a la izquierda.
- Espaciado (getbootstrap.com/docs/4.0/utilities/spacing/)
 - Las clases .p*, controlan el padding; las clases .m* controlan el margen.
 - .pb-4, establece el padding inferior (b, bottom) a 0,5 veces el espaciado definido en la hoja de estilos.
 - .pr-sm-2: en pantallas de más de 575px de viewport, el padding derecho se establece a 0,25 veces el espacio establecido.
 - .mt-4: establece el margen superior a 0,5 veces el ancho establecido.
 - .mr-auto, establece el margen derecho de forma automática.
 - .mx-auto, establece el margen izquierdo y derecho de forma automática (centra el texto).

Ejemplo (VIII)

- Color, aspecto (getbootstrap.com/docs/4.0/utilities/colors/)
 - .muted: atenúa el texto.
 - text-*, establece el color del primer plano (.text-white, lo establece como blanco).
 - .bg-*, establece el color de fondo del elemento (.bg-primary lo establece al color establecido como primario, el azul).
- Botones (getbootstrap.com/docs/4.0/components/buttons/)
 - .btn: en un elemento de tipo <a> le da un aspecto de botón.
 - .btn-light: el fondo del botón será claro.
 - .btn-outline-dark: la línea de contorno del botón es oscura.
- Barras de navegación (getbootstrap.com/docs/4.0/components/navbar/)
 - .navbar: crea la barra de navegación.
 - .navbar-dark, establece el color de la barra de navegación.
 - .navbar-expand-sm, establece el punto de ruptura dónde se colapsa la barra.
 - .navbar-toggler y .navbar-toggler-icon, define el botón de tipo «hamburger» que aparece cuando se colapsa la barra.
 - .collapse y .navbar-collapse, establecen las opciones asociadas a la barra de navegación.
 - .navbar-nav, define las opciones del menú.
 - .nav-link, define cada una de las opciones; .active, establece cuál es la opción actual.

UNIVERSIDAD
INTERNACIONAL
DE LA RIOJA

unir

www.unir.net