

1. Contenidos de Bootstrap



¿Qué es Bootstrap?

Bootstrap es un framework (librerías de CSS) que nos facilita y estandariza el desarrollo de sitios web.

A partir de la versión 3.x ha sido implementado pensando en la adaptación a diferentes tipos de dispositivos (monitores, tabletas, celulares, etc).

Bootstrap ha sido desarrollada y es mantenida por la empresa **Twitter** y la ha liberado como un producto **Open Source**.

Tiene una filosofía muy intuitiva para el maquetado de sitios web que puede ser rápidamente aprendida por desarrolladores que no vienen del mundo del diseño web.

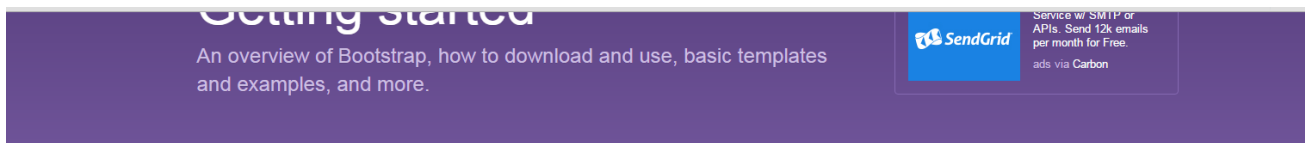
El corazón de este framework es un archivo CSS que lo podemos descargar del sitio <http://getbootstrap.com>.

Descargar Bootstrap

Existen varias formas diferentes de empezar con Bootstrap, cada una orientada a un tipo de público en función de su nivel técnico.

- **Descargar el código CSS y JavaScript compilado**, que es la forma más sencilla de empezar a utilizar Bootstrap. La desventaja es que esta versión no incluye ni los archivos originales ni la documentación. Para descargar esta versión, accede a getbootstrap.com y pulsa el botón **Download Bootstrap**.
- **Descargar el código fuente**, contiene todos los archivos Less, y JavaScript originales de Bootstrap. La desventaja es que requiere un compilador de archivos Less y cierto trabajo de configuración. Descarga la versión más reciente en el sitio github.com/twbs/bootstrap/releases.
- **Descargar el código fuente en formato Sass**, se trata de una variante de la versión anterior y que se ha creado para facilitar la integración de Bootstrap en las aplicaciones Ruby On Rails, Compass o cualquier otro proyecto basado en Sass. Descarga la versión más reciente en el sitio github.com/twbs/bootstrap-sass/releases.

Nosotros trabajaremos con la versión compilada de Bootstrap.



Download

Bootstrap (currently v3.3.6) has a few easy ways to quickly get started, each one appealing to a different skill level and use case. Read through to see what suits your particular needs.

Bootstrap

Compiled and minified CSS, JavaScript, and fonts. No docs or original source files are included.

[Download Bootstrap](#)

Source code

Source Less, JavaScript, and font files, along with our docs. **Requires a Less compiler and some setup.**

[Download source](#)

Sass

Bootstrap ported from Less to Sass for easy inclusion in Rails, Compass, or Sass-only projects.

[Download Sass](#)

[Bootstrap CDN](#)

- Download
- What's included
- Compiling CSS and JavaScript
- Basic template
- Examples
- Tools
- Community
- Disabling responsiveness
- Migrating from 2.x to 3.0
- Browser and device support
- Third party support
- Accessibility
- License FAQs
- Translations

Contenidos de la versión compilada de Bootstrap

Después de descomprimir el archivo que hemos descargado con la versión compilada de Bootstrap, veremos la siguiente estructura de archivos y directorios:

```
bootstrap/  
├── css/  
│   ├── bootstrap.css  
│   ├── bootstrap.min.css  
│   ├── bootstrap-theme.css  
│   └── bootstrap-theme.min.css  
├── js/  
│   ├── bootstrap.js  
│   └── bootstrap.min.js  
└── fonts/  
    ├── glyphicons-halflings-regular.eot  
    ├── glyphicons-halflings-regular.svg  
    ├── glyphicons-halflings-regular.ttf  
    └── glyphicons-halflings-regular.woff
```

Estos archivos son la forma más sencilla de utilizar Bootstrap en cualquier proyecto web. Para cada archivo se ofrecen dos variantes: los archivos compilados (cuyo nombre es bootstrap.*) y los archivos compilados + comprimidos (cuyo nombre es bootstrap.min.*). También se incluyen las fuentes de los iconos del proyecto Glyphicons y el tema opcional de Bootstrap.

Tipos de grillas

Diseño con plantillas

Antes de comenzar a diseñar el layout o estructura de contenidos de las páginas, es necesario realizar algunos preparativos importantes:

Doctype de HTML5

Bootstrap utiliza algunos elementos HTML y algunas propiedades CSS que requieren el uso del doctype de HTML5.

```
<!doctype html>  
  
<html lang="es">  
  
...  
  
</html>
```

Responsive design

Para que las páginas se muestren correctamente y el zoom funcione bien en los dispositivos móviles, debemos añadir la siguiente etiqueta dentro de la cabecera <head> de las páginas:

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

Si queremos deshabilitar el zoom en nuestras páginas, debemos añadir la propiedad user-scalable=no a la etiqueta anterior:

```
<meta name="viewport" content="width=device-width, initial-scale=1,  
maximum-scale=1, user-scalable=no">
```

Al añadir la propiedad user-scalable=no, los usuarios ya no podrán hacer zoom en la página y solamente podrán hacer scroll en sus contenidos. El resultado es que el comportamiento de la página se parece más al de una aplicación móvil nativa.

Imágenes responsive

Para hacer que las imágenes de nuestro contenido también sean adaptables, debemos añadir la clase `.img-responsive` a cada imagen que necesitamos que se comporte de manera responsive. Esta clase incluye las propiedades `max-width: 100%`; y `height: auto`; para que la imagen escale en función del tamaño del elemento en el que se encuentra.

```

```

Tipografía y enlaces

Bootstrap establece una serie de estilos por defecto para la tipografía de todos los elementos y para los enlaces de la página.

- Se establece a blanco el color de fondo del body con la propiedad **background-color: white**;
- Se utiliza el valor de las variables **@font-family-base**, **@font-size-base** y **@line-height-base** definidas por LESS como atributos de las propiedades tipográficas de los elementos.
- Se establece el color de los enlaces al valor de la variable **@link-color** de LESS y sólo se muestran los enlaces subrayados en el estado `:hover`
- Esta primera inicialización de estilos se define en el archivo `scaffolding.less`.

Centrando los contenidos de la página

Si queremos centrar una página respecto a la ventana del navegador, debemos encerrar sus contenidos dentro de un elemento y aplicarle la clase `.container`:

```
<div class="container">
```

```
...
```

```
</div>
```

La anchura del contenedor varía en cada punto de ruptura del diseño para adaptarse a la plantilla. Los contenedores no se pueden anidar debido a la propiedad padding y a su anchura fija.

Tipos

Bootstrap incluye una rejilla o retícula fluída pensada para móviles y que cumple con el diseño web responsive. Esta retícula crece hasta 12 columnas a medida que aumenta el tamaño de la pantalla del dispositivo. Bootstrap incluye clases CSS para utilizar la rejilla directamente en nuestros diseños y también define mixins de LESS para que podamos crear diseños más semánticos.

El diseño de páginas basado en rejilla se realiza mediante filas y columnas donde se colocan los contenidos. Así funciona la rejilla de Bootstrap:

- Las filas siempre se definen dentro de un contenedor de tipo `.container` (anchura fija) o de tipo `.container-fluid` (anchura variable). De esta forma las filas se alinean bien y muestran el padding correcto.
- Las filas se utilizan para agrupar horizontalmente a varias columnas.
- El contenido siempre se coloca dentro de las columnas, ya que las filas sólo deberían contener como hijos elementos de tipo columna.
- Bootstrap define muchas clases CSS (como por ejemplo `.row` y `.col-xs-4`) para crear rejillas rápidamente.
- La separación entre columnas se realiza aplicando padding. Para contrarrestar sus efectos en la primera y última columnas, las filas (elementos `.row`) aplican márgenes negativos.
- Debemos utilizar la clase "row" para indicar el comienzo de una fila. Dentro de la fila dispondremos tantas cajas como columnas tenga dicha fila.
- Las columnas de la rejilla definen su anchura especificando cuántas de las 12 columnas de la fila ocupan. Si por ejemplo queremos dividir una fila en tres columnas iguales, utilizarías la clase `.col-xs-4` (el 4 indica que cada columna ocupa 4 de las 12 columnas en las que se divide cada fila).

Media queries

Bootstrap utiliza las siguientes media queries para establecer los diferentes puntos de ruptura en los que la rejilla se transforma para adaptarse a cada dispositivo.

```
/* Dispositivos muy pequeños (teléfonos de hasta 768px de anchura) */
```

```
/* No se define ninguna media query porque este es el estilo por defecto utilizado por Bootstrap 4 */
```

```
/* Dispositivos pequeños (tablets, anchura mayor o igual a 576px) */
```

```
@media (min-width: @screen-sm-min) { ... }
```

```
/* Dispositivos medianos (ordenadores, anchura mayor o igual a 768px) */
```

```
@media (min-width: @screen-md-min) { ... }
```

```
/* Dispositivos grandes (ordenadores, anchura mayor o igual a 992px) */
```

```
@media (min-width: @screen-lg-min) { ... }
```

```
/* Dispositivos extra grandes (ordenadores, anchura mayor o igual a 1200px) */
```

```
@media (min-width: @screen-xl-min) { ... }
```

	Extra small <576px	<small>Recomendado</small> Small ≥576px	Medium ≥768px	Large ≥992px	Extra large ≥1200px
Max container width	None (auto)	540px	720px	960px	1140px
Class prefix	.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-
# of columns	12				
Gutter width	30px (15px on each side of a column)				
Nestable	Yes				
Column ordering	Yes				

Ejemplo de rejilla creada con Bootstrap

El siguiente ejemplo muestra cómo crear una rejilla con las clases .col-md-*. En los dispositivos móviles esta rejilla se muestra verticalmente, pero en una computadora se ve horizontalmente.

Ver ejemplo_grid.html

Alto de columna

Como las rejillas de Bootstrap tienen cinco puntos de ruptura en los que las columnas se reordenan, es probable que tengamos problemas cuando las columnas tengan diferente altura. Para solucionarlo, debemos utilizar la clase .clearfix combinándola con alguna de las clases auxiliares tipo .visible-xs:

```
<div class="row">
  <div class="col-xs-6 col-sm-3">.col-xs-6 .col-sm-3</div>
  <div class="col-xs-6 col-sm-3">.col-xs-6 .col-sm-3</div>
```

```
<!-- La clase 'clearfix' sólo se aplica cuando el dispositivo sea
      muy pequeño, tal y como indica la clase 'visible-xs' -->
```



```
<div class="clearfix visible-xs"></div>

<div class="col-xs-6 col-sm-3">.col-xs-6 .col-sm-3</div>

<div class="col-xs-6 col-sm-3">.col-xs-6 .col-sm-3</div>

</div>
```

También es posible que en ocasiones necesitemos resetear los desplazamientos de las columnas. Las clases que resetean estos valores sólo están disponibles para los dispositivos medianos y grandes, que los desplazamientos sólo funcionan en esos dispositivos.

```
<div class="row">

  <div class="col-sm-5 col-md-6">.col-sm-5 .col-md-6</div>

  <div class="col-sm-5 col-sm-offset-2 col-md-6 col-md-offset-0">.col-sm-5
  .col-sm-offset-2 .col-md-6 .col-md-offset-0</div>

</div>
```

```
<div class="row">

  <div class="col-sm-6 col-md-5 col-lg-6">.col-sm-6 .col-md-5 .col-lg-6</div>

  <div class="col-sm-6 col-md-5 col-md-offset-2 col-lg-6 col-lg-offset-0">.col-sm-6
  .col-md-5 .col-md-offset-2 .col-lg-6 .col-lg-offset-0</div>

</div>
```

Desplazando columnas

Para desplazar cualquier columna hacia su derecha, debemos añadir la clase `.col-md-offset-*`. Estas clases aumentan el tamaño del margen izquierdo de la columna en una cantidad equivalente a esas *columnas. La clase `.col-md-offset-4` por ejemplo desplaza la columna una anchura equivalente a 4columnas.

```
<div class="row">
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4 col-md-offset-4">.col-md-4 .col-md-offset-4</div>
</div>
```

```
<div class="row">
  <div class="col-md-3 col-md-offset-3">.col-md-3 .col-md-offset-3</div>
  <div class="col-md-3 col-md-offset-3">.col-md-3 .col-md-offset-3</div>
</div>
```

```
<div class="row">
  <div class="col-md-6 col-md-offset-3">.col-md-6 .col-md-offset-3</div>
</div>
```

Anidando columnas

Bootstrap 3 también permite anidar columnas dentro de otras columnas. Para ello, dentro de una columna con la clase `col-md-*` creamos un nuevo elemento con la clase `.row` y añadimos una o más columnas con la clase `.col-md-*`. Las columnas anidadas siempre tienen que sumar 12 columnas de anchura, tal y como muestra el siguiente ejemplo.

```
<div class="row">
  <div class="col-md-9">
    Level 1: .col-md-9
    <div class="row">
      <div class="col-md-6">
        Level 2: .col-md-6
```

```
</div>  
  
<div class="col-md-6">  
  Level 2: .col-md-6  
</div>  
  
</div>  
  
</div>  
  
</div>
```

Variables y mixins de LESS

Además de las clases CSS listas para definir rejillas rápidamente, Bootstrap incluye variables y mixins de LESS para generar fácilmente tus propios diseños web semánticos.

Variables

Las variables establecen el número de columnas, su separación y la anchura del navegador a partir de la cual las columnas flotan horizontalmente en vez de mostrarse una encima de otra. Los valores por defecto de estas variables son los que se muestran a continuación:

```
@grid-columns:          12;  
@grid-gutter-width:     30px;  
@grid-float-breakpoint: 768px;
```

Mixins

Los mixins, junto con las variables anteriores, permiten crear estilos semánticos para los diferentes elementos de la rejilla.

```
// Crea un elemento contenedor de varias columnas  
  
.make-row(@gutter: @grid-gutter-width) {  
  // Limpiar las columnas flotadas  
  .clearfix();  
  
  @media (min-width: @screen-small) {  
    margin-left: (@gutter / -2);  
    margin-right: (@gutter / -2);  
  }  
}
```

```
// Aplicar un margen negativo a la fila para alinear el
// contenido de las columnas
.row {
  margin-left: (@gutter / -2);
  margin-right: (@gutter / -2);
}
}

// Generar las columnas extra pequeñas
.make-xs-column(@columns; @gutter: @grid-gutter-width) {
  position: relative;
  // Evitar que las columnas no se vean cuando están vacías
  min-height: 1px;
  // Utilizar padding para separar las columnas
  padding-left: (@gutter / 2);
  padding-right: (@gutter / 2);

  // Calcular la anchura en función del número de columnas
  @media (min-width: @grid-float-breakpoint) {
    float: left;
    width: percentage((@columns / @grid-columns));
  }
}
```

```
// Generar las columnas pequeñas

.make-sm-column(@columns; @gutter: @grid-gutter-width) {
    position: relative;

    // Evitar que las columnas no se vean cuando están vacías
    min-height: 1px;

    // Utilizar padding para separar las columnas
    padding-left: (@gutter / 2);
    padding-right: (@gutter / 2);

    // Calcular la anchura en función del número de columnas
    @media (min-width: @screen-small) {
        float: left;
        width: percentage((@columns / @grid-columns));
    }
}

// Generate the small column offsets
.make-sm-column-offset(@columns) {
    @media (min-width: @screen-small) {
        margin-left: percentage((@columns / @grid-columns));
    }
}

.make-sm-column-push(@columns) {
```

```
@media (min-width: @screen-small) {
    left: percentage((@columns / @grid-columns));
}
}

.make-sm-column-pull(@columns) {
    @media (min-width: @screen-small) {
        right: percentage((@columns / @grid-columns));
    }
}

// Generar las columnas medianas

.make-md-column(@columns; @gutter: @grid-gutter-width) {
    position: relative;
    // Evitar que las columnas no se vean cuando están vacías
    min-height: 1px;
    // Utilizar padding para separar las columnas
    padding-left: (@gutter / 2);
    padding-right: (@gutter / 2);

    // Calcular la anchura en función del número de columnas
    @media (min-width: @screen-medium) {
        float: left;
        width: percentage((@columns / @grid-columns));
    }
}
```

```
}

// Generar los desplazamientos de las columnas medianas
.make-md-column-offset(@columns) {
  @media (min-width: @screen-medium) {
    margin-left: percentage((@columns / @grid-columns));
  }
}

.make-md-column-push(@columns) {
  @media (min-width: @screen-medium) {
    left: percentage((@columns / @grid-columns));
  }
}

.make-md-column-pull(@columns) {
  @media (min-width: @screen-medium) {
    right: percentage((@columns / @grid-columns));
  }
}

// Generar las columnas grandes
.make-lg-column(@columns; @gutter: @grid-gutter-width) {
  position: relative;

  // Evitar que las columnas no se vean cuando están vacías
  min-height: 1px;
}
```



```
// Utilizar padding para separar las columnas
padding-left: (@gutter / 2);
padding-right: (@gutter / 2);

// Calcular la anchura en función del número de columnas
@media (min-width: @screen-large) {
  float: left;
  width: percentage((@columns / @grid-columns));
}
}

// Generar los desplazamientos de las columnas grandes
.make-lg-column-offset(@columns) {
  @media (min-width: @screen-large) {
    margin-left: percentage((@columns / @grid-columns));
  }
}

.make-lg-column-push(@columns) {
  @media (min-width: @screen-large) {
    left: percentage((@columns / @grid-columns));
  }
}

.make-lg-column-pull(@columns) {
```

```
    right: percentage((@columns / @grid-columns));  
  }  
}
```

Implementación

Utilizando los mixins anteriores y modificando el valor de las variables para ajustarlos a nuestras necesidades, ya podemos crear diseños web semánticos. Este ejemplo muestra cómo crear un diseño a dos columnas con una separación entre los dos:

```
.wrapper {  
  .make-row();  
}  
  
.content-main {  
  .make-column(8);  
}  
  
.content-secondary {  
  .make-column(3);  
  .make-column-offset(1);  
}  
  
<div class="wrapper">  
  <div class="content-main">...</div>  
  <div class="content-secondary">...</div>  
</div>
```

Tipografía

Los estilos relacionados con la tipografía y el texto de los contenidos son esenciales en cualquier framework CSS. Por esa razón, Bootstrap 3 incluye decenas de estilos para los principales elementos utilizados en los sitios y aplicaciones web.

Títulos

Bootstrap define estilos por defecto para todos los niveles de titulares de las páginas, desde <h1>hasta <h6>.

h1. Bootstrap heading Semibold 36px

h2. Bootstrap heading Semibold 30px

h3. Bootstrap heading Semibold 24px

h4. Bootstrap heading Semibold 18px

h5. Bootstrap heading Semibold 14px

h6. Bootstrap heading Semibold 12px

Texto

El tamaño de letra (font-size) por defecto de Bootstrap 3 es 14px y el interlineado (line-height) es 1.428. Estos valores se aplican tanto al <body> como a todos los párrafos. Estos últimos también incluyen un margen inferior cuyo valor es la mitad que su interlineado (10px por defecto).

Nullam quis risus eget urna mollis ornare vel eu leo. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Nullam id dolor id nibh ultricies vehicula.

Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec ullamcorper nulla non metus auctor fringilla. Duis mollis, est non commodo luctus, nisi erat porttitor ligula, eget lacinia odio sem nec elit. Donec ullamcorper nulla non metus auctor fringilla.

Maecenas sed diam eget risus varius blandit sit amet non magna. Donec id elit non mi porta gravida at eget metus. Duis mollis, est non commodo luctus, nisi erat porttitor ligula, eget lacinia odio sem nec elit.

Clases CSS

Clases CSS para alinear texto

Bootstrap 3 define varias clases CSS para alinear de diferentes formas el contenido de texto de los elementos.

Ejemplo:

```
<p class="text-left">Texto alineado a la izquierda.</p>
```

```
<p class="text-center">Texto centrado.</p>
```

```
<p class="text-right">Texto alineado a la derecha.</p>
```

Texto alineado a la izquierda.

Texto centrado.

Texto alineado a la derecha.

Clases CSS para indicar el tipo de contenido

Bootstrap 3 también define varias clases CSS para mostrar el tipo de contenido a través del color del texto.

Ejemplo:

```
<p class="text-muted">Fusce dapibus, tellus ac cursus commodo, tortor mauris nibh.</p>
```

```
<p class="text-primary">Nullam id dolor id nibh ultricies vehicula ut id elit.</p>
```

```
<p class="text-success">Duis mollis, est non commodo luctus, nisi erat porttitor ligula.</p>
```

```
<p class="text-info">Maecenas sed diam eget risus varius blandit sit amet non magna.</p>
```

```
<p class="text-warning">Etiam porta sem malesuada magna mollis euismod.</p>
```

```
<p class="text-danger">Donec ullamcorper nulla non metus auctor fringilla.</p>
```

Fusce dapibus, tellus ac cursus commodo, tortor mauris nibh.

Nullam id dolor id nibh ultricies vehicula ut id elit.

Duis mollis, est non commodo luctus, nisi erat porttitor ligula.

Maecenas sed diam eget risus varius blandit sit amet non magna.

Etiam porta sem malesuada magna mollis euismod.

Donec ullamcorper nulla non metus auctor fringilla.

Listas

Listas no ordenadas

Bootstrap 3 también incluye los estilos adecuados para las las listas anidadas.

- Lorem ipsum dolor sit amet
- Consectetur adipiscing elit
- Integer molestie lorem at massa
- Facilisis in pretium nisl aliquet
- Nulla volutpat aliquam velit
 - Phasellus iaculis neque
 - Purus sodales ultricies
 - Vestibulum laoreet porttitor sem
 - Ac tristique libero volutpat at
- Faucibus porta lacus fringilla vel
- Aenean sit amet erat nunc
- Eget porttitor lorem

Listas ordenadas

1. Lorem ipsum dolor sit amet
2. Consectetur adipiscing elit
3. Integer molestie lorem at massa
4. Facilisis in pretium nisl aliquet
5. Nulla volutpat aliquam velit
6. Faucibus porta lacus fringilla vel
7. Aenean sit amet erat nunc
8. Eget porttitor lorem

Listas sin estilo

Como resulta muy habitual mostrar las listas sin viñetas y sin margen izquierdo, Bootstrap 3 incluye una clase CSS llamada **.list-unstyled** para aplicar esos estilos.

Lorem ipsum dolor sit amet
Consectetur adipiscing elit
Integer molestie lorem at massa
Facilisis in pretium nisl aliquet
Nulla volutpat aliquam velit
◦ Phasellus iaculis neque
◦ Purus sodales ultricies
◦ Vestibulum laoreet porttitor sem
◦ Ac tristique libero volutpat at
Faucibus porta lacus fringilla vel
Aenean sit amet erat nunc
Eget porttitor lorem

Listas en línea

También resulta habitual mostrar los elementos de una lista horizontalmente, como por ejemplo en el menú principal de navegación. Para ello, Bootstrap 3 define una clase CSS llamada **.inline-block**.

Lorem ipsum Phasellus iaculis Nulla volutpat

Listas de definición

No se utilizan mucho, pero Bootstrap 3 también incluye estilos por defecto para las listas de definición, formadas por pares término + definición.

Description lists

A description list is perfect for defining terms.

Euismod

Vestibulum id ligula porta felis euismod semper eget lacinia odio sem nec elit.

Donec id elit non mi porta gravida at eget metus.

Malesuada porta

Etiam porta sem malesuada magna mollis euismod.

Listas de definición horizontales

Si lo preferimos, también es posible mostrar la lista de definición horizontalmente añadiendo la clase **.dl-horizontal**.

Description lists	A description list is perfect for defining terms.
Euismod	Vestibulum id ligula porta felis euismod semper eget lacinia odio sem nec elit. Donec id elit non mi porta gravida at eget metus.
Malesuada porta	Etiam porta sem malesuada magna mollis euismod.
Felis euismod sempe...	Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet risus.

Elementos CSS

Bootstrap 3 define estilos para las tablas y las imágenes, incluyendo todas sus variantes. También se explican las clases CSS genéricas y utilidades que simplifican el diseño de los sitios web y que se pueden aplicar a cualquier elemento.

Tablas

Tablas básicas

Con la clase `.table` añadida a cualquier elemento `<table>` podemos aplicar los estilos básicos de Bootstrap 3 para tablas. El resultado es una tabla con un padding muy sutil y con líneas de separación solamente en las filas.

Ejemplo:

```
<table class="table">
```

```
...
```

```
</table>
```

#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

Tablas cebreadas

Las tablas cebreadas son aquellas cuyas filas alternan su color de fondo para mejorar la legibilidad de los contenidos. En Bootstrap 3 estas tablas se crean añadiendo la clase `.table-striped`.

Ejemplo:

```
<table class="table table-striped">
```

...

```
</table>
```

#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

Tablas con bordes

Si queremos utilizar el estilo tradicional de las tablas con los cuatro bordes en todas las celdas y en la propia tabla, tenemos que añadir la clase `.table-bordered`.

Ejemplo:

```
<table class="table table-bordered">
```

...

```
</table>
```

#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

Tablas responsive

La solución que propone Bootstrap 3 para crear tablas responsive que se vean bien en dispositivos pequeños consiste en añadir un scroll horizontal a las tablas que sean demasiado anchas. Para ello, debemos enmarcar cualquier tabla con la

clase `.table` dentro de un elemento con la clase `.table-responsive`. Cuando las tablas responsive se muestran en dispositivos con una anchura superior a 768px, se ven igual que cualquier otra tabla normal.

Ejemplo:

```
<div class="table-responsive">
  <table class="table">
    ...
  </table>
</div>
```

#	Cabecera de tabla	Cabecera de tabla	C
1	Celda de tabla	Celda de tabla	C
2	Celda de tabla	Celda de tabla	C
3	Celda de tabla	Celda de tabla	C

Imágenes

Bootstrap 3 define varias clases CSS para las imágenes:

- `.img-rounded`, añade unas pequeñas esquinas redondeadas en todos los lados de la imagen aplicando el estilo `border-radius: 6px`.
- `.img-thumbnail`, muestra la imagen con un relleno blanco y un borde fino simulando el aspecto de las fotografías de las antiguas cámaras instantáneas. Añade además una breve animación para hacer que la imagen aparezca al cargar la página.
- `.img-circle`, convierte la imagen en un círculo aplicando el estilo `border-radius: 50%`



Utilidades responsive

Las utilidades responsive permiten diseñar con mayor facilidad sitios web móviles, ya que muestran u ocultan elementos en función del tipo de dispositivo que utiliza el usuario para navegar. También se incluyen clases para mostrar/ocultar elementos al imprimir la página.

Estas clases deben utilizarse con moderación y siempre para mejorar el aspecto de los contenidos en cada tipo de dispositivo. Además, sólo funcionan para los elementos de bloque y las tablas.

Podemos utilizar alguna de estas clases o combinar varias entre sí para definir cómo se ven los contenidos en cada tipo de dispositivo:

- Celular = menos de 768px;
- Tableta = más de 768px;
- Computadora = más de 992px;
- Computadora grande = más de 1200px.

Clase	Celulares	Tabletas	Computadoras	Comp. Grande
.visible-xs	Visible	Oculto	Oculto	Oculto
.visible-sm	Oculto	Visible	Oculto	Oculto
.visible-md	Oculto	Oculto	Visible	Oculto
.visible-lg	Oculto	Oculto	Oculto	Visible
.hidden-xs	Oculto	Visible	Visible	Visible

.hidden-sm	Visible	Oculto	Visible	Visible
.hidden-md	Visible	Visible	Oculto	Visible
.hidden-lg	Visible	Visible	Visible	Oculto