

Introducción a Bootstrap

- Por qué Utilizar Bootstrap.....3
- Cómo Crear una Plantilla con Bootstrap.....4
- Configuración de Bootstrap.....4
- Diseñando el menú.....5
- Diseñando la cabecera.....8
- Sistema de rejilla (grid).....9
- Iconos en Bootstrap.....11
- Personaliza los Estilos de Bootstrap.....13

Introducción a Bootstrap

Bootstrap, sin lugar a dudas es el framework HTML, CSS y JS más utilizado. Bootstrap permite la creación de proyectos web responsivos que se adaptan a todo tipo de dispositivos, incluyendo una gran cantidad de componentes que usan estilos prediseñados.

La [documentación de Bootstrap](#) es excelente, incluyendo todo lo que necesitas tanto para diseñar una web como para gestionar diferentes integraciones con herramientas de desarrollo. Podrás incluir directamente los recursos de bootstrap en tu web o también instalarlo mediante algún gestor de paquetes como Bower o [npm](#). También podrás encontrar el código de Bootstrap en sus versiones LESS y Sass, de modo que cualquier cambio en alguno de sus componentes por defecto sea muy sencillo.

Lo cierto es que hace años no era habitual el uso de frameworks CSS, siendo algo que requería cierta pericia por parte del desarrollador al igual que ocurre con cualquier nueva tecnología. Sin embargo, a día de hoy ocurre totalmente lo contrario, siendo lo habitual el uso de este tipo de frameworks incluso en el caso de un desarrollador sin demasiada experiencia. Esto quizás sea debido a que hoy en día es de esperar que cualquier aplicación sea responsiva, siendo mucho más sencillo el uso de un framework que aprender en detalle todos los conceptos que permiten lograr este tipo de diseño.

Este tutorial de Bootstrap pretende ser introductorio, por lo que no entraremos en conceptos propios de preprocesadores CSS como LESS o [Sass](#). Entre otras cosas, también aprenderás lo que es un Framework CSS y cómo puede ayudarte, además de diversos métodos de incluir y utilizar los archivos CSS y JS de Bootstrap en tu proyecto. No necesitarás ningún conocimiento previo salvo ciertos conocimientos básicos de HTML y [CSS](#)

Qué es Bootstrap

Bootstrap es un framework CSS, JS y HTML que permite crear webs responsivas. Incluye un archivo **.css**, un archivo **.js** y, en versiones previas a la 4, también una serie de iconos llamados **glyphicons**. Además, también utiliza jQuery, que como quizás ya sepas es una famosa librería de JavaScript compatible con todos los navegadores que facilita y estandariza varias tareas.

Tal y como verás en su documentación, podrás instalar Bootstrap mediante gestores de paquetes como npm o Bower. Del mismo modo también te podrás descargar Bootstrap en su versión Sass o LESS. Sin embargo esto no es en absoluto necesario y está fuera del alcance de ese tutorial, así que en caso de que hayas leído su documentación y veas que estos conceptos aparecen por todas partes, continúa sin miedo.

Por qué Utilizar Bootstrap

Lo cierto es que, al igual que ocurre con cualquier tecnología, un framework nunca es necesario, si bien es cierto que a la hora de la verdad te ahorrará mucho tiempo y quebraderos de cabeza. Con Bootstrap evitarás realizar una y otra vez ciertas tareas repetitivas. Además su diseño es de raíz **responsivo**, por lo que los contenidos de tu web se adaptarán de forma nativa a cualquier tipo de pantalla, ya sea la pantalla de un teléfono, una **tablet** o un monitor de escritorio. Otra de las ventajas que obtendrás con **Bootstrap** es un diseño consistente en todos los elementos visuales de tu proyecto, funcionando correctamente con todos los navegadores, siendo posible también la creación de prototipos con gran rapidez.

También existen otras muchas ventajas de **Bootstrap**, y es que al ser un **framework open source** dispone de una gran comunidad trabajando en él que identifica y soluciona posibles problemas. Por otro lado proporciona también una metodología estándar de diseño, útil cuando se trabaja en equipo. Es decir, es prácticamente imposible que un desarrollador recién llegado a una empresa conozca un **framework** que solamente se utiliza en esa empresa, pero es muy probable que ya tenga experiencia con **Bootstrap**. Esto da lugar a una arquitectura visual y a un diseño más consistentes.

Si por algo destaca **bootstrap** es por proporcionar una **grid o rejilla que permite dividir la pantalla en una serie de filas y columnas que se adaptarán a cualquier tipo de pantalla**. Además también agrega estilos a formularios, botones, listas, imágenes y a muchos otros componentes habituales.

.col-md-8		.col-md-4
.col-md-4	.col-md-4	.col-md-4
.col-md-6	.col-md-6	

NeoGuías

También incluye diseños CSS predefinidos junto con otros elementos más complejos como puede ser una barra de **menús de navegación**. Mediante el uso de JavaScript también se hace posible el uso de muchos elementos, como **popups**, **ventanas emergentes**, menús o listas interactivas de elementos con forma de acordeón que se abren o se cierran cuando haces clic en alguno de sus elementos.

Cómo Crear una Plantilla con Bootstrap

En la web oficial de Bootstrap podrás encontrar [innumerables ejemplos](#) acerca de cómo crear un tema usando los elementos y estilos nativos que incluye. Lo que haremos en este apartado es crear un tema básico usando estos estilos y componentes y, tras ello, modificaremos dichos estilos para crear un tema personalizado, ya que no suele ser demasiado atractivo el uso del mismo tema en todos y cada uno de tus proyectos.

Configuración de Bootstrap

Lo primero que haremos será [descargar Bootstrap desde su web oficial](#). El archivo comprimido que te descargarás incluye en su interior directorios que contienen archivos CSS y archivos JS. Debes descargarte la versión que incluye los recursos CSS y los archivos JavaScript ya compilados.

Vamos a crear un directorio para este tutorial de demostración. En mi caso le llamaré **/tutorial-bootstrap** y lo situaré en mi carpeta de proyectos. Luego crea un directorio llamado **/css** y otro llamado **/js** en su interior.

Seguidamente abre el archivo zip que te has descargado y copia el archivo **bootstrap.min.css** en el directorio **/css** del proyecto. Del mismo modo, copia el archivo **bootstrap.min.js** en la carpeta **/js** del proyecto.

Ahora crea un archivo llamado **index.html** en la carpeta raíz del proyecto e incluye el siguiente código, que no es más que una [plantilla básica HTML](#):

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="utf-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <title>Tutorial de Bootstrap</title>
  <link href="css/bootstrap.min.css" rel="stylesheet" />
</head>
<body>
  <h1>El mejor tutorial de Bootstrap!</h1>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="js/bootstrap.min.js"></script>
```

```
</body>
</html>
```

Con esto ya tenemos una plantilla básica desde la que empezar a trabajar. Tenemos la etiqueta `<doctype>` y el código en el interior de la etiqueta `<html>`, incluyendo la cabecera `<head>`, el cuerpo principal del archivo en el `<body>` y todas las metaetiquetas `<meta>` necesarias, entre las que se encuentra el `<viewport>`, una etiqueta importante en el **diseño responsivo**, ya que nos asegura un ratio de visualización de 1:1 con respecto al tamaño de la pantalla del usuario.

Tal y como puedes ver, hemos agregado el archivo `.css` de bootstrap en la sección `<head>`:

```
<link href="css/bootstrap.min.css" rel="stylesheet" />
```

Justo antes del fin de la etiqueta `</body>` también hemos agregado el archivo `.js` de Bootstrap y también **jQuery**, necesario para que Bootstrap funcione:

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="js/bootstrap.min.js"></script>
```

Es importante que incluyas jQuery antes que Bootstrap. Del mismo modo, cualquier script que incluyas que haga uso de jQuery también debe incluirse después. En este caso hemos cargado jQuery desde los servidores de Google, pero también puedes [descargarlo desde aquí](#) e incluirlo desde tu directorio de scripts si lo prefieres.

Si abres el archivo `index.html` en tu navegador, por ahora, deberías ver algo como esto:

El mejor tutorial de Bootstrap!

Diseñando el menú

Vamos a ver cómo agregar una barra de menús a la página. En este enlace vemos [cómo crear una barra de menús responsiva](#) en otro tutorial, cosa que nos llevará cierto tiempo. Sin embargo, **Bootstrap** ya incluye un menú por defecto. Para incluirlo, copia y pega el siguiente menú de ejemplo justo después de la etiqueta `<body>`:

```

<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
  <a class="navbar-brand" href="#">Tutorial</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbar-menu" aria-controls="navbar-menu" aria-expanded="false" aria-
label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
  <div class="collapse navbar-collapse" id="navbar-menu">
    <ul class="navbar-nav mr-auto">
      <li class="nav-item active">
        <a class="nav-link" href="#">Inicio<span class="sr-only">(actual)</span></a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Enlace</a>
      </li>
      <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button"
data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
          Expandir
        </a>
        <div class="dropdown-menu" aria-labelledby="navbarDropdown">
          <a class="dropdown-item" href="#">Enlace</a>
          <a class="dropdown-item" href="#">Otro enlace</a>
        </div>
      </li>
    </ul>
  </div>
</nav>

```

Lo que hemos hecho ha sido lo siguiente. La etiqueta **nav** de Bootstrap incluye por defecto los estilos de un menú. En este caso hemos aplicado las clases **navbar-dark** y **bg-dark** para que su aspecto sea oscuro en lugar de claro. En caso de querer que la barra menús siempre estuviese en la parte superior incluso cuando haces scroll hacia abajo, tendrías que agregar la clase **fixed-top**.

En el interior del menú hemos definido un enlace con el nombre de nuestra página, que en este caso es el de Tutorial, aunque en esta sección también es habitual incluir una imagen con el logo de tu web:

```
<a class="navbar-brand" href="#">Tutorial</a>
```

Seguidamente hemos incluido un botón mediante el cual podremos abrir o cerrar el menú en dispositivos móviles. Este botón solamente se mostrará en pantallas pequeñas:

```
<button class="navbar-toggler" type="button" data-toggle="collapse" data-  
target="#navbar-menu" aria-controls="navbar-menu" aria-expanded="false" aria-  
label="Toggle navigation">  
  <span class="navbar-toggler-icon"></span>  
</button>
```

Mediante el atributo **data-target** del botón podremos controlar el elemento que se desplegará, que en este caso es el elemento con id **navbar-menu** correspondiente a nuestro menú. Este botón es similar al creado en [este tutorial](#), a diferencia de que el botón de Bootstrap usa el típico icono con forma de hamburguesa, por lo que no está animado. En caso de que quieras mejorarlo, basta con que sigas el tutorial anterior.

Seguidamente hemos agregado el menú con id **navbar-menu** y los elementos del menú en sí:

```
<div class="collapse navbar-collapse" id="navbar-menu">  
  <ul class="navbar-nav mr-auto">  
    <li class="nav-item active">  
      <a class="nav-link" href="#">Inicio<span class="sr-only">(actual)</span></a>  
    </li>  
    <li class="nav-item">  
      <a class="nav-link" href="#">Enlace</a>  
    </li>  
    <li class="nav-item dropdown">  
      <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button"  
data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">  
        Expandir  
      </a>  
      <div class="dropdown-menu" aria-labelledby="navbarDropdown">  
        <a class="dropdown-item" href="#">Enlace</a>
```

```

    <a class="dropdown-item" href="#">Otro enlace</a>
  </div>
</li>
</ul>
</div>

```

Aquí tienes el resultado de lo que hemos hecho hasta ahora en su versión móvil:



Diseñando la cabecera

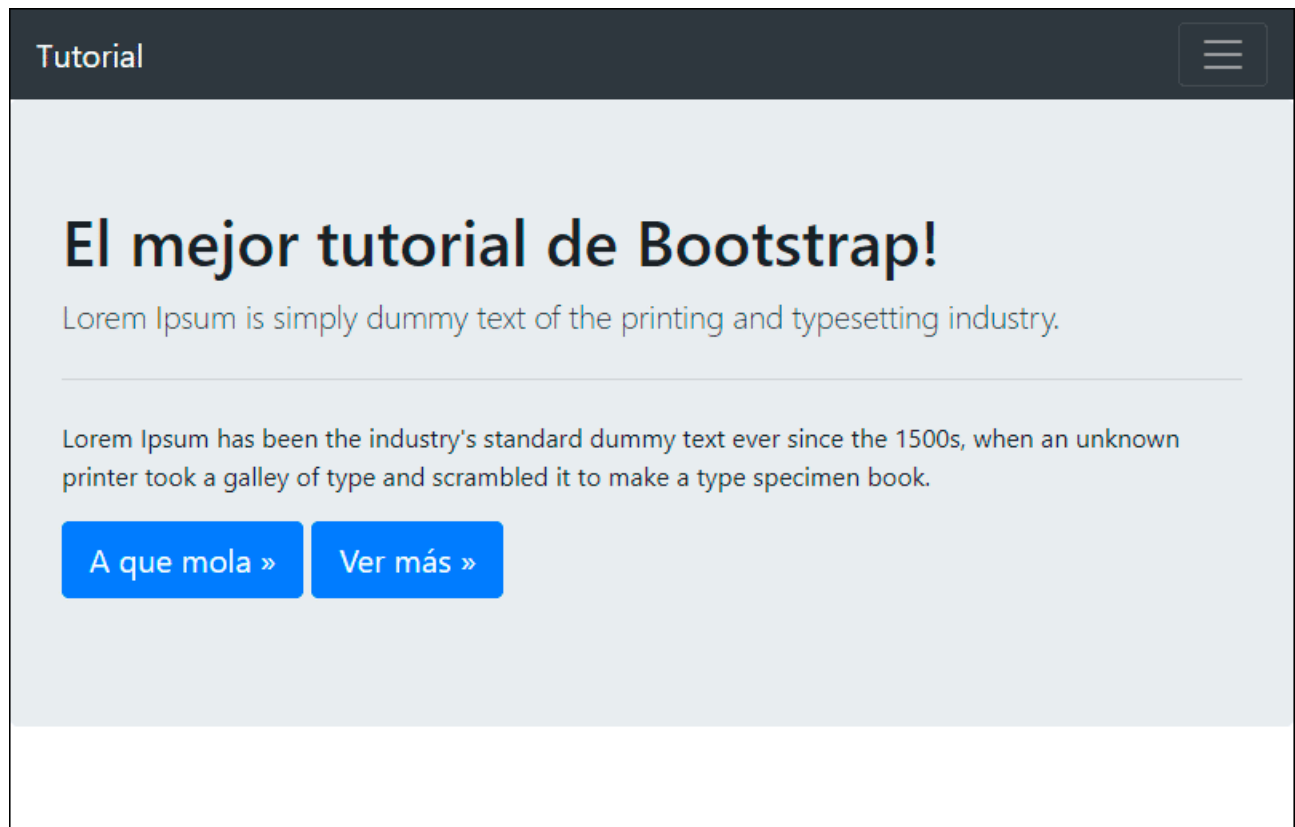
Ahora vamos a crear una cabecera en la parte superior de la página que llame algo la atención. Para ello vamos a usar el elemento **jumbotron** de Bootstrap:

```

<div class="jumbotron">
  <h1>El mejor tutorial de Bootstrap!</h1>
  <p class="lead">
    Lorem Ipsum is simply dummy text of the printing and typesetting industry.
  </p>
  <hr class="my-4">
  <p>
    Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when
    an unknown printer took a galley of type and scrambled it to make a type specimen book.
  </p>
  <p class="lead">
    <a class="btn btn-primary btn-lg" href="#" role="button">A que mola »</a>
    <a class="btn btn-primary btn-lg" href="#" role="button">Ver más »</a>
  </p>
</div>

```


Además hemos movido el título `<h1>` de sitio para que esté en el interior del **jumbotron**. Este es el resultado de lo que hemos hecho hasta ahora:



Sistema de rejilla (grid)

Ha llegado la hora de agregar algunos contenidos. Para organizarlos utilizaremos la grid o rejilla que viene por defecto con Bootstrap, de modo que podamos organizar los contenidos en una serie de filas y columnas responsivas. Esta rejilla es similar a la [grid con flexbox de este tutorial](#), aunque en este caso cada fila se divide por defecto en 12 columnas.

Para empezar, en Bootstrap existen una serie de contenedores o **containers** que nos permiten definir filas en su interior. Podemos definir un **container** de este modo.

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

Estos **containers** tienen un ancho máximo que por defecto es de 1200px. Sin embargo, podemos definir también contenedores fluidos sin ancho máximo, de modo que siempre ocupen la totalidad del ancho de la ventana del elemento dentro del cual estén incluidos:

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

Dentro de cada **container** podremos agregar contenidos o definir todas las filas que queramos. El objetivo de las filas es el de contener una serie de columnas en su interior:

```
<div class="row"> </div>
```

Dentro de cada fila podrás definir hasta doce columnas de anchos diferentes. El espacio que ocupa cada columna se define mediante una clase CSS que tendrá cada una. En el siguiente ejemplo vamos a definir una fila que contenga dos columnas en su interior, de forma que cada una ocupe un 50% de la fila:

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

Como ves, la clase **col-md-x** se usa para definir el espacio que ocupa cada columna, siendo **x** el espacio de cada una, que puede ir desde 1 hasta un total de 12, sumando el total de las columnas un máximo de 12, que representa el 100% del ancho.

En caso de querer definir tres columnas, tendrías que hacer lo siguiente:

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

Lo que vamos a agregar a nuestro proyecto es un **container** que contenga una fila con tres columnas, de modo que podamos crear una serie de características destacadas de aquello que estemos creando:

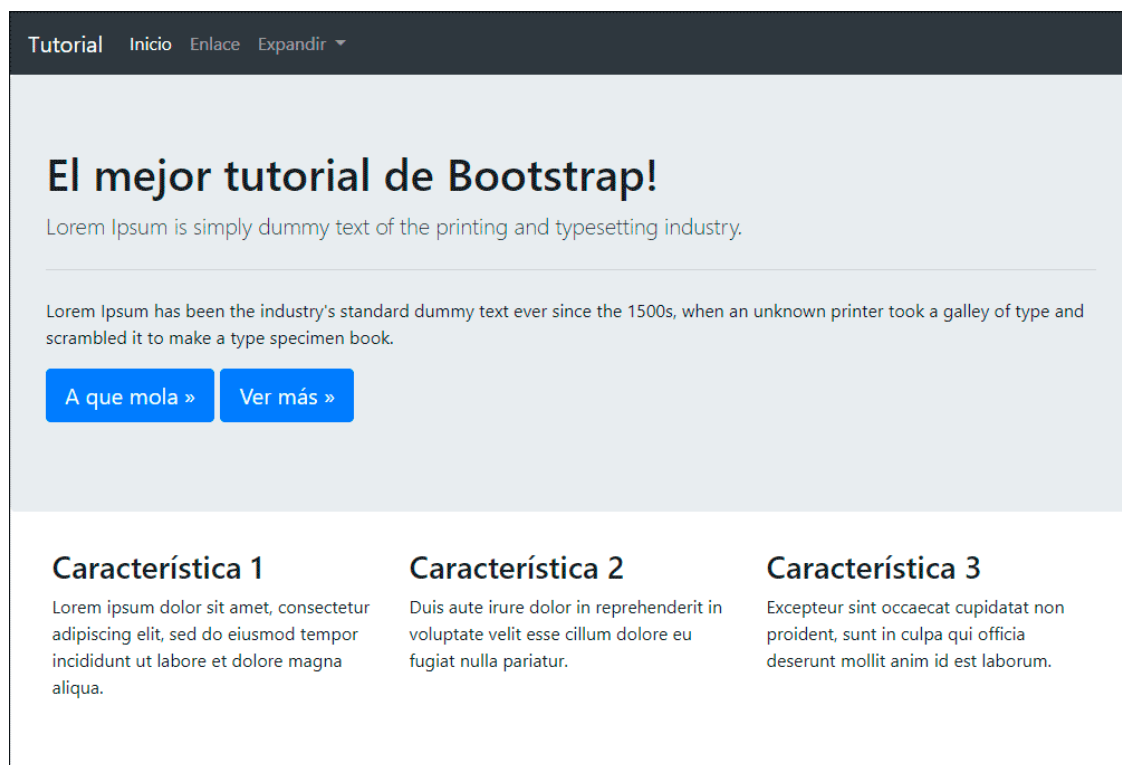
```
<div class="container">
  <div class="row">
    <div class="col-md-4">
      <h3>Característica 1</h3>
      <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
    </div>
    <div class="col-md-4">
      <h3>Característica 2</h3>
      <p>Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.</p>
    </div>
  </div>
</div>
```

```

<div class="col-md-4">
  <h3>Característica 3</h3>
  <p>Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</p>
</div>
</div>
</div>

```

Este es el resultado de lo que hemos hecho hasta ahora:



Iconos en Bootstrap

Tampoco estaría mal si agregamos una serie de iconos a las características destacadas del proyecto. Bootstrap incluía antes los famosos glyphs, que son una serie de iconos que se incluían con el framework. Sin embargo se descartó su uso en la versión 4 Bootstrap, por lo que tendremos que incluir algún pack de iconos. El conjunto de iconos más utilizado es **Font Awesome**, que incluye una versión gratuita con bastantes iconos que podremos integrar en el proyecto.

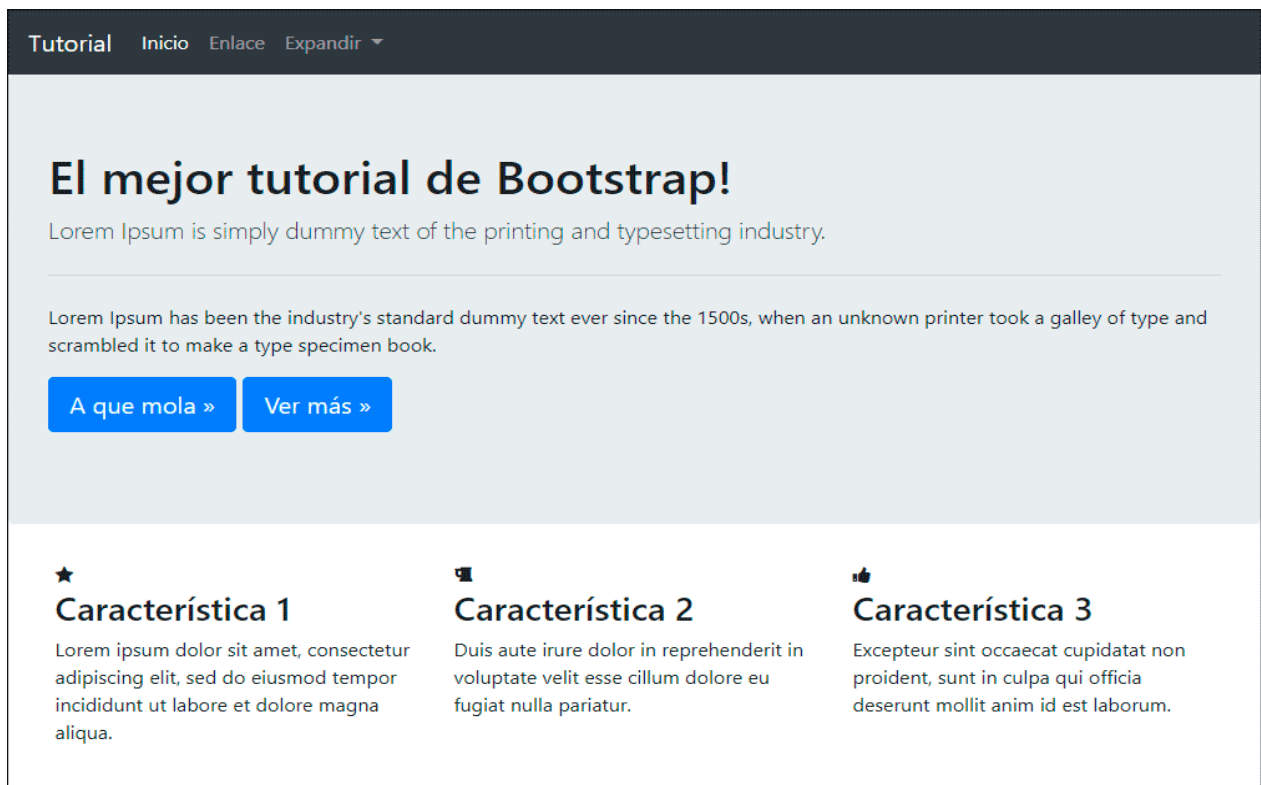
Para integrar Font Awesome incluye la siguiente hoja de estilos en la cabecera del proyecto:

```
<link href="https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css" rel="stylesheet">
```

Ahora vamos a agregar los iconos, a los que les daremos un estilo más tarde. No agregues todo este código, sino que solamente debes modificar el anterior agregando los iconos:

```
<div class="container">
  <div class="row">
    <div class="col-md-4">
      <i class="fa fa-star"></i>
      <h3>Característica 1</h3>
      <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
    </div>
    <div class="col-md-4">
      <i class="fa fa-beer"></i>
      <h3>Característica 2</h3>
      <p>Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.</p>
    </div>
    <div class="col-md-4">
      <i class="fa fa-thumbs-up"></i>
      <h3>Característica 3</h3>
      <p>Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</p>
    </div>
  </div>
</div>
```

Aquí tienes el aspecto de lo que hemos hecho por ahora:



Personaliza los Estilos de Bootstrap

Por ahora hemos obtenido un diseño más que aceptable y responsivo que se ve bien en cualquier tipo de dispositivo. Sin embargo existen miles de webs así, siendo de todo menos un diseño original, por lo que es hora de agregar algunos cambios.

En la web de bootstrap puedes encontrar varios temas y, además, también puedes realizar cambios mediante LESS o SASS para crear una versión personalizada. Otra opción consiste en crear una hoja de estilos CSS que agregue estilos adicionales que es lo que vamos a hacer. Para ello vamos a crear un archivo llamado **estilos.css** en el directorio **/css** de nuestro proyecto. Una vez crees el archivo, agrégalo a la cabecera del proyecto, justo debajo de Bootstrap:

```
<link href="css/estilos.css" rel="stylesheet" />
```

Seguidamente edita el archivo **estilos.css**, ya que vamos a agregar algunos estilos. Para empezar, vamos a cambiar el fondo de la página y del jumbotron. También vamos a modificar los encabezados, los párrafo y el color del texto:

```
body {  
  background: #3e4649;  
  color: #f7f7f7;
```

```
font-family: sans-serif;
}

h1,
h2 {
  font-weight: bold;
}

p {
  font-size: 18px;
  color: #cdcdcd;
}

.jumbotron {
  background: #e48632;
  color: white;
  text-align: center;
}

.jumbotron p {
  color: white;
  font-size: 26px;
}
```

Ahora vamos a hacer que los botones de la cabecera sean transparentes, modificando también su aspecto cuando posamos el ratón encima de ellos:

```
.btn-primary {
  color: #fff;
  background-color: transparent;
  border-color: white;
  margin-bottom: 5px;
}

.btn-primary:hover {
```

```
color: #e48632;
background-color: white;
border-color: white;
}
```

Vamos a cambiar también el aspecto del menú, eliminando el **padding** del contenedor principal y agregándolo a cada elemento, modificando también el color de fondo de cada botón cuando se posa el ratón sobre él:

```
.navbar-dark {
  padding: .0rem 10px;
  background: #2e2f31;
  border: 0;
}

.navbar-dark .navbar-nav li a {
  padding: 20px;
  color: #f7f7f7;
  font-size: 16px;
}

.navbar-dark .navbar-nav > li > a:hover {
  background: #e48632;
}

.dropdown-menu {
  background: #2e2f31;
  border-radius: 0;
  border: 0;
}

.dropdown-menu li a {
  padding: 10px;
}

.navbar-dark .navbar-nav .dropdown-menu a:hover {
  background: #3e4649 !important;
}
```

```
}
```

Ahora vamos a cambiar el estilo de los iconos, modificando su posición y haciéndolos más grandes. Para ello vamos a agregar al container en el que está (en donde también están las filas y columnas) la clase **features**. De este modo ya podemos centrar el contenido y personalizar tanto los iconos como el texto de esta sección:

```
.features {  
  text-align: center;  
}  
  
.features p {  
  margin-bottom: 30px;  
  font-family: sans-serif;  
}  
  
.features i {  
  font-size: 100px;  
}
```

Finalmente vamos a agregar una media query para modificar el padding del menú cuando el ancho de la pantalla es menor a 992px:

```
@media screen and (max-width: 991px) {  
  nav {  
    padding: 10px !important;  
  }  
}
```

Y este es el resultado final de este proyecto:

El mejor tutorial de Bootstrap!

Lorem Ipsum is simply dummy text of the printing and typesetting industry.

Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

A que mola »

Ver más »



Característica 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.



Característica 2

Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.



Característica 3

Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Como ves, hemos dado un aspecto diferente aplicando muy pocos estilos. Obviamente las posibilidades de Bootstrap van mucho más allá, aunque esto ha sido todo en esta introducción. Si no sabes cómo realizar alguna cosa, seguramente encuentres lo que buscas buscándolo en Google, ya que Bootstrap es un framework utilizado por montones de desarrolladores.

Si quieres, puedes descargar el proyecto desde el siguiente enlace:

•**Código del proyecto:** [Repositorio del tutorial de Bootstrap en GitHub](#)