

EJEMPLO DE CUADRÍCULA FLUIDA CON BOOTSTRAP 5



A diferencia de otras soluciones como las vistas hasta ahora basadas exclusivamente en CSS, Bootstrap reúne los elementos prefabricados necesarios para construir un sitio web con contenido fluido y *responsive*. Se ha desecho de dependencias que arrastraban versiones anteriores siendo la principal esa atadura a la librería jQuery. Para ello sus ingenieros han reescrito desde cero el código de cada componente BS5 en Javascript plano (Vanilla JS), mejorando mucho el rendimiento así como las peculiaridades de cada navegador.

Como era de esperar, sigue el paradigma *mobile first* y sus cuadrículas son tanto **responsive** (achicando y recolocando contenidos) como **fluidas** (similar aspecto, tamaño proporcional en función del viewport) según se requiera. Esta tecnología basa conceptualmente la cuadrícula en una **división en 12 columnas**, teniendo predefinidos contenedores tanto fluidos como de ancho fijo. Los *media queries* se encargan de establecer los estilos para varios tamaños diferentes de pantalla utilizando una nomenclatura como **sm** (small), **md** (medium), **lg** (large), **xl** (extralarge) y **xxl** (extraextralarge).

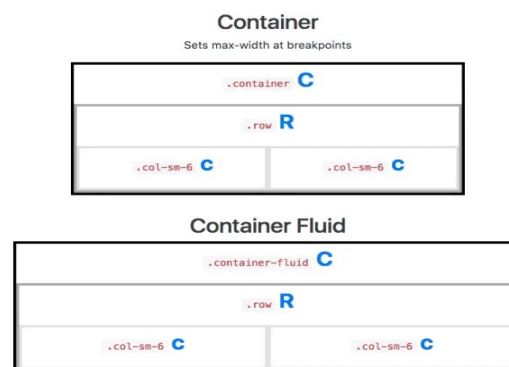
Breakpoint	Prefijo Bootstrap	Resolución
X-Small	Sin prefijo	Menos de 576px
Small	sm	>= 576px
Medium	md	>= 768px
Large	lg	>= 992px
Extra large	xl	>= 1200px
Extra extra large	xxl	>= 1400px

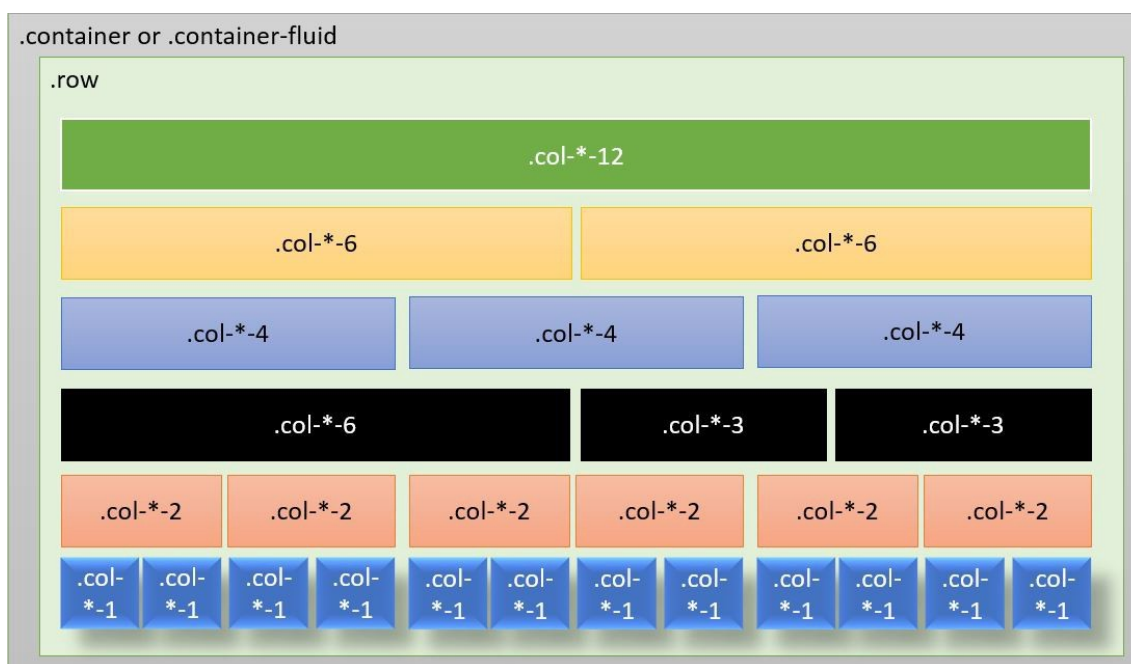
La cuadrícula se subdividirá en una o más filas (según el desarrollador vaya necesitando), cada una de las cuales usará la clase **row**. El objetivo de este contenido de segundo nivel es efectuar los **ajustes necesarios en los márgenes** de los elementos, a fin de lograr una perfecta alineación del contenido.

Cada fila estará dividida en las secciones que interesen, cada una de las cuales puede ocupar **entre 1 y 12 columnas de ancho**.

Los selectores CSS siguen el patrón **col-tipo-N**, siendo **tipo** uno de esos sm/md/lg/xl vistos en el cuadro anterior y **N** el número de columnas que se quiere ocupar. Cualquiera de estas unidades puede contener, a su vez, un elemento de tipo **container** (para *responsive*) o **container-fluid** (para fluidas), facilitando así la composición de distribuciones anidadas.

Abundando en lo anterior, la ilustración de abajo nos va conectando la idea de nuestro diseño a los códigos que deberemos emplear en Bootstrap para conseguirlo:





Los asteriscos que aparecen después de `.col-` se corresponderán con el tipo asociado a cada resolución donde rompa el tamaño de pantalla establecido (sm/md/lg/xl/xxl). Si se nos presentara una de estas notaciones como `col-12`, deberemos interpretar lo que indica la primera fila del cuadro de Tipo y Resolución de la página anterior: **sin prefijo** → **pantallas de menos de 576 píxeles**.

Preparando Bootstrap

Ya hemos indicado con anterioridad que Bootstrap 5 se ha librado para siempre de la librería jQuery, pero eso no significa que Javascript no sea necesario. Es más, se señaló que los componentes estaban reescritos en Javascript puro y duro. Eso implica entonces que cualquier código BS5 que incluyamos en los documentos ha de localizar el prefabricado que le corresponde.

Todo ese código prefabricado en CSS y JS lo ha reunido Bootstrap en un fichero de forma que para ser utilizado en nuestro proyecto nos lo podemos montar de dos maneras:

a) si nos vale con la configuración por defecto, es cuestión de enlazar directamente los módulos CSS y JavaScript de Bootstrap a través del CDN correspondiente. Esta vía es la más recomendable a la hora de agregar Bootstrap a un proyecto dado que proporciona un beneficio de rendimiento al reducir el tiempo de carga del documento. Cuando un usuario solicite dicho fichero, al estar en múltiples servidores lo recibirá desde el más cercano. En la actualidad, para la versión 5 se está utilizando este CDN (aunque hay que consultar la página de Bootstrap para ir sabiendo las nuevas versiones que van saliendo):

```
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/
bootstrap.min.css" rel="stylesheet" integrity="sha384-
1BmE4kWBq78iYhFIdvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBo
qyl2QvZ6jIW3" crossorigin="anonymous" />
```

b) si la configuración por defecto se nos queda corta y buscamos un punto mayor de personalización, entonces hay que descargar ese fichero **bootstrap.min.css** a nuestro equipo y, una vez retocado, alojarlo en el mismo servidor que el sitio web.

Hay que señalar también que algunos componentes Bootstrap requieren algo más de código Javascript, de ahí que para que puedan funcionar para nuestro proyecto debemos tirar de un segundo fichero llamado **bundle** (.js, de ahí la etiqueta script):

```
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/
bootstrap.bundle.min.js" integrity="sha384-
ka7Sk0GlIn4gmtz2MlQnikT1wXgYsOg+OMhuP+IIRH9sENBOOL
Rn5q+8nbTov4+1p" crossorigin="anonymous"></script>
```

Documento de partida

Así quedaría de inicio nuestro documento al prepararlo para la utilización de Bootstrap mediante CDN:

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Cuadrícula con Bootstrap 5</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<!-- Bootstrap 5 mediante CDN -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-
1BmE4kWBq78iYhFdvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous" />
<style>
p { padding: 1em; border: 1px solid gray }
.multi { background: orange }
.single { background: lime }
</style>
</head>
<body>
<h1>Cuadrícula flexible con Bootstrap5 mediante CDN</h1>
</body>
</html>
```

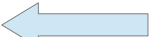
(el fichero del AV se denomina **1a.CuadrículaBootstrap.html**)



Pero también se puede descargar (nosotros ya lo hemos hecho y está en /css) y enlazar como si de una hoja de estilos CSS de toda la vida se tratara:

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Cuadrícula con Bootstrap 4</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<!-- Bootstrap 5 mediante fichero ubicado en local -->
<link href="css/bootstrap.min.css" rel="stylesheet">
```

(el fichero del AV se denomina **1b.CuadrículaBootstrap.html**)



```

<style>
  p { padding: 1em; border: 1px solid gray }
  .multi { background: orange}
  .single { background: lime }
</style>
</head>
<body>
  <h1>Cuadrícula flexible con Bootstrap5 mediante fichero descargado</h1>
</body>
</html>

```

A partir de ahora continuaremos alternando la **opción de fichero descargado** y la opción de **enlace CDN**.

¡Importante!, antes de continuar

Como el 99,99% de los elementos en Bootstrap 5 están prefabricados, es muy complicado para cualquier humano tener en la cabeza el nombre de cada pieza prefabricada. Es por ello que Bootstrap ha elaborado un sitio web con toda la documentación que se quiera consultar todo por categorías y fácilmente accesible. La versión con la que estamos trabajando es la 5.1 y el lugar para consultarlo todo sobre ella está en [<https://getbootstrap.com/docs/5.1/getting-started/introduction/>].

Preparando el marco

Ya se indicó en un momento anterior que para incluir elementos contenedores de tipo *responsive* había que tirar de la propiedad **container**. Pues dicho y hecho:

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Cuadrícula con Bootstrap 5</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!-- Bootstrap 5 mediante fichero ubicado en local -->
    <link href="css/bootstrap.min.css" rel="stylesheet">
    <style>
      p { padding: 1em; border: 1px solid gray }
      .multi { background: orange}
      .single { background: lime }
    </style>
  </head>
  <body>
    <h1>Cuadrícula flexible con Bootstrap5 mediante fichero descargado</h1>
    <div class="container">
  </div>
</body>
</html>

```

(el fichero del AV se denomina **2a.CuadrículaBootstrap.html**)

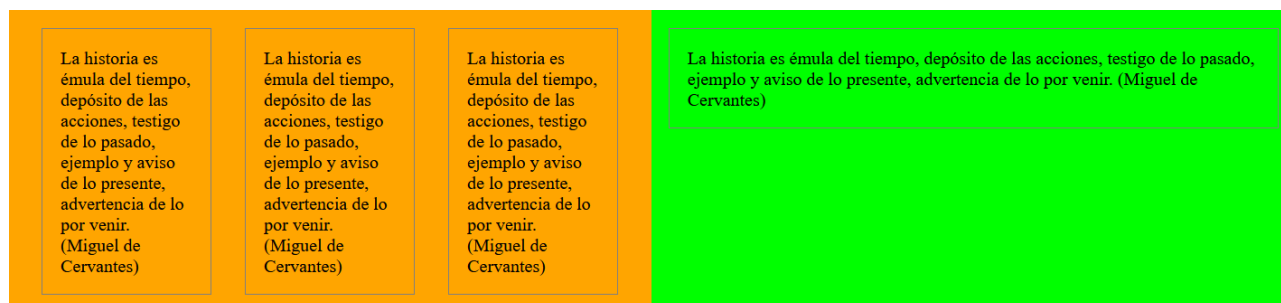


Pero antes de continuar hay marcar una pincelada de concepto ya que nos vamos a encontrar en nuestro deambular por Bootstrap con contenedores por defecto (clase container), contenedores fluidos (clase container-fluid) y contenedores responsivos (clases container-sm/-md/-lg/-xl/-xxl). La mejor explicación de las diferencias entre ellos la encontrarás en [el vídeo del amigo Kiko Palomares](#) que utiliza la siguiente tabla tan clarificadora:

	Extra pequeño <576px	Small ≥576px	Medium ≥768px	Large ≥992px	X-Large ≥1200px	XX-Large ≥1400px
.container	100%	540px	720px	960px	1140px	1320px
.container-sm	100%	540px	720px	960px	1140px	1320px
.container-md	100%	100%	720px	960px	1140px	1320px
.container-lg	100%	100%	100%	960px	1140px	1320px
.container-xl	100%	100%	100%	100%	1140px	1320px
.container-xxl	100%	100%	100%	100%	100%	1320px
.container-fluid	100%	100%	100%	100%	100%	100%

Preparando las columnas principales

Ahora viene el momento de acordarse de las propiedades **row** y **cols** ya vistas y adecuarlas al diseño que pretendemos hacer:



Como vemos, en la captura realizada para pantalla de portátil, las dos partes que componen el documento ocuparán un tamaño similar. Luego razonando, y teniendo en cuenta las 12 columnas en las que se basa Bootstrap, dejaremos 6 para una y las 6 restantes para la otra:

<code><div class="col-12 col-md-8 col-lg-6 multi"></code>	<code><div class="col-12 col-md-4 col-lg-6 single"></code>
---	--

Pero como debemos estar preparados para que el contenido pueda ser visualizado en pantallas más pequeñas, le indicamos que para pantallas medianas, la parte de la izquierda pueda ocupar dos tercios (**col-md-8**) y la de la derecha pueda quedarse con un tercio (**col-md-4**). Cuando ya nos vayamos a pantallas pequeñas de verdad, tipo smartphone, le indicamos que lo ocupen todo el tamaño que pueden (**col-12**).

```

<!DOCTYPE html>
<html lang="en">
<head>
  <title>Cuadrícula con Bootstrap 5</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- Bootstrap 5 mediante fichero ubicado en local -->
  <link href="css/bootstrap.min.css" rel="stylesheet">

  <style>
    p { padding: 1em; border: 1px solid gray }
    .multi { background: orange }
    .single { background: lime }
  </style>

</head>
<body>
  <h1>Cuadrícula flexible con Bootstrap5 mediante fichero descargado</h1>
  <div class="container">
    <div class="row">
      <div class="col-12 col-md-8 col-lg-6 multi"> ←
    </div>
    <div class="col-12 col-md-4 col-lg-6 single"> ←
  </div>
</div>
</body>
</html>

```

(el fichero sigue siendo **2a.CuadrículaBootstrap.html**)

Para tener una segunda visión del funcionamiento de filas y columnas en Bootstrap 5, puedes recurrir a [este vídeo de Kiko Palomares](#)

Preparando las columnas interiores

En el diseño observamos cómo la parte de la izquierda contiene dentro de la columna principal tres subcolumnas. La derecha es simple por lo que no requerirá mucho más trabajo.

Siguiendo con el ejemplo con el que estábamos trabajando, nos centramos en la columna de la izquierda en la que deberemos insertar un nuevo contenedor. Este sólo lo queremos fluido por lo que tiraremos de la propiedad **container-fluid** al que, obviamente, le sucederá la propiedad **row**. Para cada subcolumna se ha establecido que en las pantallas medianas ocupen cada una un tercio del espacio (**col-md-4**) y en caso de las pequeñas se expandan al completo (**col-12**).

```

<!DOCTYPE html>
<html lang="en">
<head>
  <title>Cuadrícula con Bootstrap 5</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- Bootstrap mediante fichero ubicado en local -->
  <link href="css/bootstrap.min.css" rel="stylesheet">
  <style>
    p { padding: 1em; border: 1px solid gray }
    .multi { background: orange }

```

(el fichero sigue siendo **2a.CuadrículaBootstrap.html**)

```

    .single { background: lime }
  </style>
</head>
<body>
  <h1>Cuadrícula flexible con Bootstrap4 mediante fichero descargado</h1>
  <div class="container">
    <div class="row">
      <div class="col-12 col-md-8 col-lg-6 multi">
        <div class="container-fluid">
          <div class="row">
            <div class="col-12 col-md-4">
              <p>La historia es émula del tiempo, depósito de las acciones, testigo de lo pasado, ejemplo y aviso de lo presente, advertencia de lo por venir. (Miguel de Cervantes)</p>
            </div>
            <div class="col-12 col-md-4">
              <p>La historia es émula del tiempo, depósito de las acciones, testigo de lo pasado, ejemplo y aviso de lo presente, advertencia de lo por venir. (Miguel de Cervantes)</p>
            </div>
            <div class="col-12 col-md-4">
              <p>La historia es émula del tiempo, depósito de las acciones, testigo de lo pasado, ejemplo y aviso de lo presente, advertencia de lo por venir. (Miguel de Cervantes)</p>
            </div>
          </div>
        </div>
      </div>
      <div class="col-12 col-md-4 col-lg-6 single">
        <p>La historia es émula del tiempo, depósito de las acciones, testigo de lo pasado, ejemplo y aviso de lo presente, advertencia de lo por venir. (Miguel de Cervantes)</p>
      </div>
    </div>
  </div>
</body>
</html>

```

(el fichero **2b.CuadrículaBootstrap.html** contiene la versión CDN de esta solución)

Un ejemplo de formulario en vídeo

Puedes seguir los pasos de este ejemplo para la confección de un formulario muy básico con Bootstrap 5, atendiendo [a este vídeo de Kiko Palomares](#)

Fíjate bien en cómo dimensiona las columnas según lo requiere la solución que piden y cómo aplica tanto las separaciones entre componentes como los centrados verticales de las etiquetas.

Un ejemplo más completo

Ahora vamos a implementar un documento que nos vaya empezando a parecer más real a los trabajos que nos tocará hacer en un futuro. En este caso un formulario *responsive* algo más resultón que el anterior en el que hemos optado por hacerlo vía CDN, que se vaya adaptando a los diferentes anchos de pantalla. Si lo seguimos correctamente debe quedarnos un resultado como el siguiente:

Formulario de Lenguaje

Lenguaje

Descripción

Número de Programadores

Guardar

Limpiar

Listado de lenguajes

#	Lenguaje	Cantidad de programadores
1	JavaScript	21
2	Python	110
3	C#	45
4	Ruby	31

Observa cómo el documento está dividido en dos partes horizontales: a la izquierda el formulario y a la derecha la lista de lo que se va insertando en el formulario. Pero ambas partes no son iguales. La de la izquierda coge un tercio del tamaño de pantalla, dejando los otros dos tercios para el resto. Eso ya nos dice algo y debemos reflejarlo en el código.

1) Esto sería el código base para comenzar cualquier proyecto con Bootstrap 5.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Formulario con Bootstrap 5</title>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- Bootstrap 5 vía CDN -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-1BmE4kWBq78iYhFIdvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
    crossorigin="anonymous" />
</head>
<body>
</body>
</html>
```

2) Ahora hay que añadirle el espaciado a izquierda y derecha. Analizando esto, podemos aplicar una fila (row) que dentro tendrá una primera columna para el formulario que ocupará 4 partes de las 12 totales (un tercio) siendo las 8 restantes para la segunda columna que albergará la tabla.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Formulario con Bootstrap 5</title>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- Bootstrap 5 vía CDN -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-1BmE4kWBq78iYhFIdvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
    crossorigin="anonymous" />
```



```

</head>
<body>
  <div class="container">
    <div class="row">
      <div class="col-sm-12 col-md-4 col-lg-4 col-xl-4">

    </div>
    <div class="col-sm-12 col-md-8 col-lg-8 col-xl-8">

    </div>
  </div>
</body>
</html>

```

Salvo para pantallas pequeñas donde la tabla se situará debajo del formulario, le decimos que para el resto de pantallas ocupen un tercio y dos tercios respectivamente.

3) Toca añadir formulario y tabla. Empezaremos por esta última.

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Formulario con Bootstrap 5</title>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!-- Bootstrap 5 via CDN -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-1BmE4kWBq78iYhFIdvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
    crossorigin="anonymous" />
  </head>
  <body>
    <div class="container">
      <div class="row">
        <div class="col-sm-12 col-md-4 col-lg-4 col-xl-4">

        </div>
        <div class="col-sm-12 col-md-8 col-lg-8 col-xl-8">
          <table>
            <thead>
              <tr>
                <th>#</th>
                <th>Lenguaje</th>
                <th>Cantidad de programadores</th>
              </tr>
            </thead>
            <tbody>
              <tr>
                <td>1</td>
                <td>JavaScript</td>
                <td>21</td>
              </tr>
              <tr>
                <td>2</td>
                <td>Python</td>
                <td>110</td>
              </tr>
              <tr>
                <td>3</td>

```

```

        <td>C#</td>
        <td>45</td>
      </tr>
    </tr>
    <tr>
      <td>4</td>
      <td>Ruby</td>
      <td>31</td>
    </tr>
  </tbody>
</table>
</div>
</div>
</div>
</body>
</html>

```

Si pruebas esto en el navegador sale poco lucido.

4) La magia de Bootstrap empieza a verse ahora. Vamos a tunear la tabla con un poquito de código.

```

<!DOCTYPE html>
<html lang="en">
<head>
  <title>Formulario con Bootstrap 5</title>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- Bootstrap 5 via CDN -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
  integrity="sha384-1BmE4kWBq78iYhFIdvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
  crossorigin="anonymous" />
  <!-- Nuestro CSS -->
  <link rel="stylesheet" href="css/formulariotabla.css" />
</head>
<body>
  <div class="container my-3">
    <div class="row">
      <div class="col-sm-12 col-md-4 col-lg-4 col-xl-4">
        <div>
          <div class="col-sm-12 col-md-8 col-lg-8 col-xl-8">
            <h2>Listado de lenguajes</h2>
            <table class="table">
              <thead>
                <tr>
                  <th class="centrado">#</th>
                  <th class="centrado">Lenguaje</th>
                  <th class="centrado">Cantidad de programadores</th>
                </tr>
              </thead>
              <tbody>
                <tr>
                  <td class="centrado">1</td>
                  <td class="centrado">JavaScript</td>
                  <td class="centrado">21</td>
                </tr>
                <tr>
                  <td class="centrado">2</td>
                  <td class="centrado">Python</td>

```

Si pones `<div class="container">` en vez de `<div class="container my-3">` observa cómo aplica un margen de 3 en vertical

Si pones `<table class="table table-dark table-striped">` en vez de `<table class="table">` observa el cambio

El estilo **centrado** nos lo hacemos nosotros a manita y lo guardamos en el fichero **formulariotabla.css**:
.centrado {text-align:center; vertical-align:middle;}

```

        <td class="centrado">110</td>
      </tr>
    <tr>
      <td class="centrado">3</td>
      <td class="centrado">C#</td>
      <td class="centrado">45</td>
    </tr>
    <tr>
      <td class="centrado">4</td>
      <td class="centrado">Ruby</td>
      <td class="centrado">31</td>
    </tr>
  </tbody>
</table>
</div>
</div>
</div>
</body>
</html>

```

5) Continuamos con el formulario hasta que quede como en la ilustración inicial. Ya tenemos experiencia confeccionando formularios por lo que no nos debe resultar complicado aplicar este código.

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Formulario con Bootstrap 5</title>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!-- Bootstrap 5 via CDN -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-1BmE4kWBq78iYhFdvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jlW3"
    crossorigin="anonymous" />
    <!-- Nuestro CSS -->
    <link rel="stylesheet" href="css/formulariotabla.css" />
  </head>
  <body>
    <div class="container my-3">
      <div class="row">
        <div class="col-sm-12 col-md-4 col-lg-4 col-xl-4">
          <h2>Formulario de Lenguaje</h2>
          <form>
            <div>
              <label for="nombre">Lenguaje</label>
              <input type="text" id="nombre" placeholder="Lenguaje" autofocus />
            </div>
            <div>
              <label for="descripcion">Descripción</label>
              <textarea id="descripcion" placeholder="Descripción del lenguaje" rows="3"></textarea>
            </div>
            <div>
              <label for="programadores">Número de Programadores</label>
              <input type="number" id="programadores" />
            </div>
          </form>

```

```

        <div>
            <button>Guardar</button>
            <button>Limpiar</button>
        </div>
    </form>
</div>
<div class="col-sm-12 col-md-8 col-lg-8 col-xl-8">
    <h2>Listado de lenguajes</h2>
    <table class="table">
        <thead>
            <tr>
                <th class="centrado">#</th>
                <th class="centrado">Lenguaje</th>
                <th class="centrado">Cantidad de programadores</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td class="centrado">1</td>
                <td class="centrado">JavaScript</td>
                <td class="centrado">21</td>
            </tr>
            <tr>
                <td class="centrado">2</td>
                <td class="centrado">Python</td>
                <td class="centrado">110</td>
            </tr>
            <tr>
                <td class="centrado">3</td>
                <td class="centrado">C#</td>
                <td class="centrado">45</td>
            </tr>
            <tr>
                <td class="centrado">4</td>
                <td class="centrado">Ruby</td>
                <td class="centrado">31</td>
            </tr>
        </tbody>
    </table>
</div>
</div>
</div>
</body>
</html>

```

6) Hay que aplicar de nuevo la magia de Bootstrap pero ahora al formulario.

```

<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Formulario con Bootstrap 5</title>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge" />
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <!-- Bootstrap 5 via CDN -->
        <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"

```

```

integrity="sha384-1BmE4kWBq78iYhFIdvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
crossorigin="anonymous" />
<!-- Nuestro CSS -->
<link rel="stylesheet" href="css/formulariotabla.css" />
</head>
<body>
  <div class="container my-3">
    <div class="row">
      <div class="col-sm-12 col-md-4 col-lg-4 col-xl-4">
        <h2>Formulario de Lenguaje</h2>
        <form>
          <div class="mb-3">
            <label for="nombre" class="form-label">Lenguaje</label>
            <input type="text" class="form-control" id="nombre" placeholder="Lenguaje" autofocus />
          </div>
          <div class="mb-3">
            <label for="descripcion" class="form-label">Descripción</label>
            <textarea class="form-control" id="descripcion" placeholder="Descripción del lenguaje"
rows="3"></textarea>
          </div>
          <div class="mb-3">
            <label for="programadores" class="form-label">Número de Programadores</label>
            <input type="number" class="form-control" id="programadores" />
          </div>
          <div class="d-grid gap-2">
            <button class="btn btn-success">Guardar</button>
            <button class="btn btn-secondary">Limpiar</button>
          </div>
        </form>
      </div>
      <div class="col-sm-12 col-md-8 col-lg-8 col-xl-8">
        <h2>Listado de lenguajes</h2>
        <table class="table">
          <thead>
            <tr>
              <th class="centrado">#</th>
              <th class="centrado">Lenguaje</th>
              <th class="centrado">Cantidad de programadores</th>
            </tr>
          </thead>
          <tbody>
            <tr>
              <td class="centrado">1</td>
              <td class="centrado">JavaScript</td>
              <td class="centrado">21</td>
            </tr>
            <tr>
              <td class="centrado">2</td>
              <td class="centrado">Python</td>
              <td class="centrado">110</td>
            </tr>
            <tr>
              <td class="centrado">3</td>
              <td class="centrado">C#</td>
              <td class="centrado">45</td>
            </tr>
            <tr>
              <td class="centrado">4</td>

```

Bootstrap te lo pone fácil a la hora poner margen por debajo (**margin-bottom**) de los controles de un formulario.

Bootstrap ya tiene clases como **form-label** y **form-control** para estilizar los controles de un formulario.

Bootstrap también tiene algo para los botones. Como tienen que ocupar todo el ancho, aquí hay que recurrir a los **block buttons** que Bootstrap los despacha añadiendo en el div que los recoge la **class="d-grid gap-2"**

Y dado que los botones requieren color, Bootstrap, además de la clase **btn**, tiene una serie de colores predeterminados con nombre **btn-success** es verde, por ejemplo.

```

        <td class="centrado">Ruby</td>
        <td class="centrado">31</td>
    </tr>
</tbody>
</table>
</div>
</div>
</div>
</body>
</html>

```

Por ir rematando la faena, si te fijas el fondo del documento es un degradado. Existe una web muy útil [<https://uigradients.com/>] en la que puedes escoger en un momento la muestra de degradado que más cuadre, en nuestro caso la denominada **Azur Lane**. Al escogerla, mirando arriba a la derecha se encuentra un pequeño botón con la leyenda < >. Si lo pulsamos, ¡oh!, nos proporciona al copiarlo el código CSS necesario para incorporar este gradiente al documento.

Dado que la aplicación de este coloreado ha de hacerse al cuerpo del documento, podemos optar por añadir dicho código al fichero ya utilizado antes y llamado **formulariotabla.css**. Como ves se le aplica a la marca **body** y listo.

```
.centrado {text-align:center; vertical-align:middle;}
```

```
body {background: #7F7FD5; /* fallback for old browsers */
background: -webkit-linear-gradient(to right, #91EAE4, #86A8E7, #7F7FD5); /* Chrome 10-25,
Safari 5.1-6 */
background: linear-gradient(to right, #91EAE4, #86A8E7, #7F7FD5); /* W3C, IE 10+/ Edge, Firefox
16+, Chrome 26+, Opera 12+, Safari 7+ */ }
```

Eso sí, ahora ese degradado nos va a dejar el formulario y la tabla menos resaltados que antes. Pero todo en CSS tiene remedio. Se les puede aplicar a cada columna (una la del formulario y otra la de la tabla) un fondo blanco. Y ya que estamos con nuestro fichero CSS, procedamos:

```
.centrado { text-align:center; vertical-align:middle; }
```

```
body {background: #7F7FD5; /* fallback for old browsers */
background: -webkit-linear-gradient(to right, #91EAE4, #86A8E7, #7F7FD5); /* Chrome, Safari */
background: linear-gradient(to right, #91EAE4, #86A8E7, #7F7FD5); /* W3C, IE 10+/ Edge, Firefox
16+, Chrome 26+, Opera 12+, Safari 7+ */ }
```

```
.fondoblanco { background-color: #ffffff; }
```

Esto requiere obviamente que se añada por acumulación este fondo blanco a las columnas en el HTML5:

```

....
<div class="col-sm-12 col-md-4 col-lg-4 col-xl-4 fondoblanco">
.....
<div class="col-sm-12 col-md-8 col-lg-8 col-xl-8 fondoblanco">
....

```

Un último detalle por si nadie se dio cuenta. Resulta que el área de texto cuyo id es descripcion, en su esquina inferior derecha tiene como unos puntitos grises. Esto indica que si se pone el ratón encima podemos agrandarlo a voluntad. Lo normal es que eso no deba suceder y para ello CSS tiene el mecanismo adecuado.

```
.centrado { text-align:center; vertical-align:middle; }
```

```
body {background: #7F7FD5; /* fallback for old browsers */
```

```
background: -webkit-linear-gradient(to right, #91EAE4, #86A8E7, #7F7FD5); /* Chrome, Safari */
```

```
background: linear-gradient(to right, #91EAE4, #86A8E7, #7F7FD5); /* W3C, IE 10+/ Edge, Firefox 16+, Chrome 26+, Opera 12+, Safari 7+ */ }
```

```
.fondoblanco { background-color: #ffffff; }
```

```
#descripcion { resize: none; }
```

Pues con todo esto ya tenemos nuestro formulario particular implementado. Aunque no estaría mal que le añadieses una barra de navegación en la parte superior para que quedara del estilo a como se ve en esta captura:

La imagen muestra una interfaz web con una barra de navegación superior que contiene los enlaces "Principal", "Productos" y "Precios". El contenido principal está dividido en dos secciones:

- Formulario de Lenguaje:** Incluye un campo de texto para "Lenguaje", un campo de texto para "Descripción" (con el placeholder "Descripción del lenguaje"), un campo de texto para "Número de Programadores" y dos botones: "Guardar" (verde) y "Limpiar" (gris).
- Listado de lenguajes:** Una tabla con tres columnas: "#", "Lenguaje" y "Cantidad de programadores".

#	Lenguaje	Cantidad de programadores
1	JavaScript	21
2	Python	110
3	C#	45
4	Ruby	31

Para ello te recomiendo que visites la página oficial de documentación para Bootstrap 5 que hace [referencia a las barras de navegación](#) y pruebes con alguna que "tuneándola" un poco te acabe dando el resultado que se pide.

Nota final: los estilos incluidos en el fichero **formulariotabla.css** se pueden ir convirtiendo en etiquetas Bootstrap 5 en el código HTML5 e ir quitándolas de dicho fichero. ¡Si te atreves...!

Las tarjetas se utilizan mucho

Son elementos contenedores que permiten reunir información de diverso tipo, imágenes, texto, botones, enlaces al modo en que podría presentarse un catálogo de productos.

Como es de imaginar Bootstrap 5, tiene una serie de etiquetas prefabricadas aplicables cada parte de una tarjeta como son card, card-body, card-title, card-text, etc.

Esta [página de la documentación de BS5](#) te permite probar muchas posibilidades de tarjetas con esta tecnología al suministrar los códigos de cada ejemplo. Pruébalos uno a uno en tu editor preferido. Puedes complementar esta página visitando [la oficial de Bootstrap 5](#) (en inglés) pero los ejemplos vienen acompañados de una ilustración para hacernos una idea de cómo queda.



Card Title

This is basic card with image on top, title, description and button.

Button

Nota: Estaría bien que probaras los conocimientos ya adquiridos sobre márgenes, padding, colores y demás para aplicarlos a estos ejemplos.

Lo que ocurre es que lo habitual es que estas tarjetas tengan que presentarse agrupadas tal como decíamos antes a modo de catálogo de productos. Estas mismas páginas de Bootstrap 5, dedican un buen rato en el apartado **Card layout** a presentar una serie de etiquetas prefabricadas para la agrupación de tarjetas. Aquí los ingenieros de BS5 siguen sin tener completo el cuadro de etiquetas posibles, aunque a día de hoy se proponen dos formas de lidiar con esto: con card-group o directamente con la cuadrícula Bootstrap (row y col).

El primero lo gestiona como un solo bloque con poco margen para el desarrollador ya que no hay separación implícita entre tarjetas y el tamaño de los elementos se ajusta como en Flexbox a la altura del más grande y la anchura a partes iguales quedando todo uniforme.

TARJETAS AGRUPADAS CON CARDGROUP



Primer Título

Aquí ponemos una descripción más extensa de la tarjeta para forzar que la altura de la caja se autoajuste al modo Flexbox.



Segundo Título

Aquí ponemos una descripción más cortita de la tarjeta.



Tercer Título

Ahora ya nos sobramos escribiendo en esta tarjeta para comprobar hasta qué punto se extiende la caja de texto y verificar cómo acaba perfectamente alineado sin ningún descuadre.

(el fichero del AV se denomina 4.TarjetaCaGroupBS5.html)

Lugar del pie primero

Lugar del pie segundo

Lugar del pie tercero

Sólo si se achica bastante la ventana del navegador, se observa que los elementos pasan a formar una sola columna en pantalla.

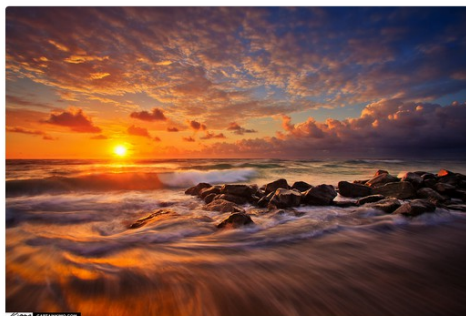
De la otra manera, al recurrir a la cuadrícula de Bootstrap, el desarrollador ya puede influir en la distribución de los elementos, por si el documento requerido exige algo más de complejidad. Eso sí, hay detalles como el de la extensión del texto en la caja que ajusta a las palabras que hay en cada tarjeta, produciendo obviamente descuadres. Ahí tendrá que intervenir el "palustre CSS" del desarrollador para afinar lo posible en función del resultado final deseado.

TARJETAS AGRUPADAS A CUADRÍCULA



Primer Título

Aquí ponemos una descripción con las suficientes palabras para que rellene una buena cantidad de espacio por debajo del título.



Segundo Título

Aquí ponemos una descripción con las suficientes palabras para que rellene una buena cantidad de espacio por debajo del título.



Tercer Título

Aquí ponemos una descripción con menos palabras para que no rellene tanto espacio.



Cuarto Título

Aquí ponemos una descripción con las suficientes palabras para que rellene una buena cantidad de espacio por debajo del título.

(el fichero del AV se denomina 5.TarjetaCuadBS5.html)

Probando su adaptabilidad en el navegador

A medida que vayas probando los códigos de esta batería en el navegador, somete a esos documentos a formatos más anchos o estrechos de ventana para observar cómo se recolocan los contenidos y comprobar que el efecto *responsive* responde a lo deseado.

Te recomiendo que pruebes los ejemplos tanto en Firefox como en Chrome.