

Desarrollo Web Frontend



Edición 2021

Universidad Autónoma de Entre Ríos
Facultad de Ciencia y Tecnología - Sede Oro Verde



¿Qué es un framework CSS?

Introducción a Bootstrap

Breakpoints

Contenedores

Grillas

Colores, botones y bordes





¿Qué es un framework CSS?

Un framework CSS es una biblioteca de estilos genéricos. Contiene utilidades que pueden ser aprovechadas permitiendo la facilidad en el diseño de sitios web.

Principales ventajas:

- Proporcionan un método rápido y fácil de implementarse en diseños web (aún cuando contamos con un diseño o maquetado existente)
- Aseguran compatibilidad en una amplia gama de navegadores
- Permite respetar estándares de diseño.
- Brindan fiabilidad ante posibles errores ya que se trata de software que está bien probado.





¿Qué es un framework CSS? (Continuación...)

Desventajas:

- Al tratarse de código prehecho existe una limitante en cuanto a posibilidades de elección del diseño web.
- Al importar el framework, es probable que exista código innecesario que no utilicemos. Lo que provoca un incremento del tiempo de carga.
- Existe la posibilidad de tener un menor control de lo que esta sucediendo por parte del diseñador. Por ejemplo al momento de querer corregir un efecto o comportamiento indeseado.

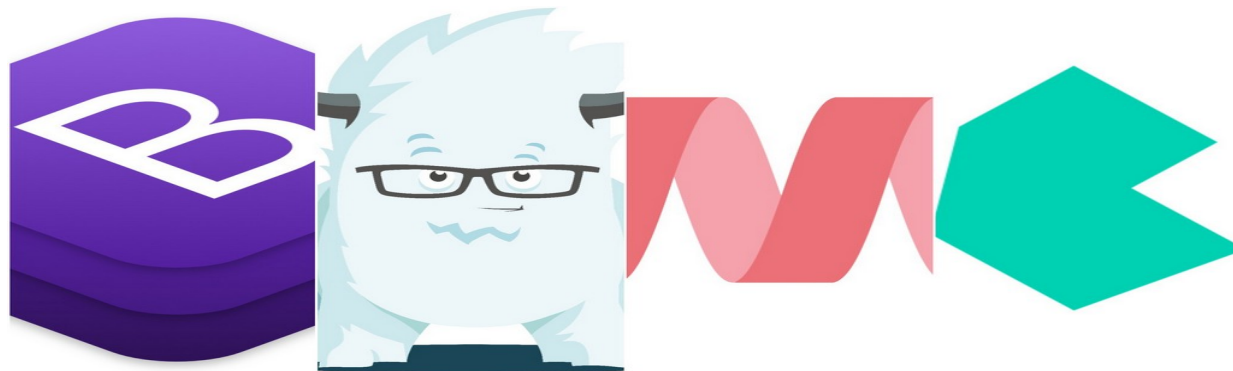




¿Qué es un framework CSS? (Continuación...)

Existen diversos frameworks CSS. Cada uno de ellos brinda un conjunto de clases y componentes para estilizar los sitio web. Entre los mas populares podemos mencionar:

- ✓ **Materialize CSS**
Framework CSS que implementa el tema de diseño **Material Design**.
- ✓ **Foundation**
Uno de los frameworks basados en componentes mas importantes del momento.
- ✓ **Bulma**
Framework basado en componentes CSS puros. No requiere Javascript para funcionar.
- ✓ **Bootstrap**
Filosofía *mobile-first*. Combina CSS y Javascript para estilizar los componentes de una página en HTML.





~~¿Qué es un framework CSS?~~

Introducción a Bootstrap

Breakpoints

Contenedores

Grillas

Colores, botones y bordes



Introducción a Bootstrap (Continuación...)

- Bootstrap es un *framework* de CSS desarrollado por Twitter en el año 2010 para estandarizar las herramientas de la Compañía
- Combina CSS y Javascript para estilizar los componentes de una pagina en HTML.
- Objetivo crear sitios web responsivos.



- Esta compuesto por 2 carpetas CSS y JS, en ellas se encuentran los archivos necesarios para estilizar los componentes y los responsables de ejecutar estilos de forma interactiva.
- Los estilos se aplican utilizando *clases* sobre los elementos HTML.
- Actualmente se encuentra en la versión 5.



Introducción a Bootstrap (Continuación...)

Proceso de instalación

- **Método 1:**

Ingresa en la sección de descargas

<https://getbootstrap.com/docs/5.0/getting-started/download/>, copiar los links de distribución de contenidos (CDN) e insertarlos dentro del **<head>** de nuestro sitio web.

CDN via jsDelivr

Skip the download with [jsDelivr](#) to deliver cached version of Bootstrap's compiled CSS and JS to your project.

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js">
```

```
<!DOCTYPE html>
<html lang="en">

<head>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
    rel="stylesheet" integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC" crossorigin="anonymous">
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
    crossorigin="anonymous"></script>
</head>
<body>
  Hola mundo!
</body>
</html>
```



Introducción a Bootstrap (Continuación...)

Proceso de instalación

- **Método 2:**

Ingresa en la sección de descargas

<https://getbootstrap.com/docs/5.0/getting-started/download/>, descargar y descomprimir el código compilado para el desarrollo dentro del directorio del proyecto. Incluir los archivos bootstrap.css y bootstrap.js dentro de las etiquetas **<head>**.

Download

[View on GitHub](#)

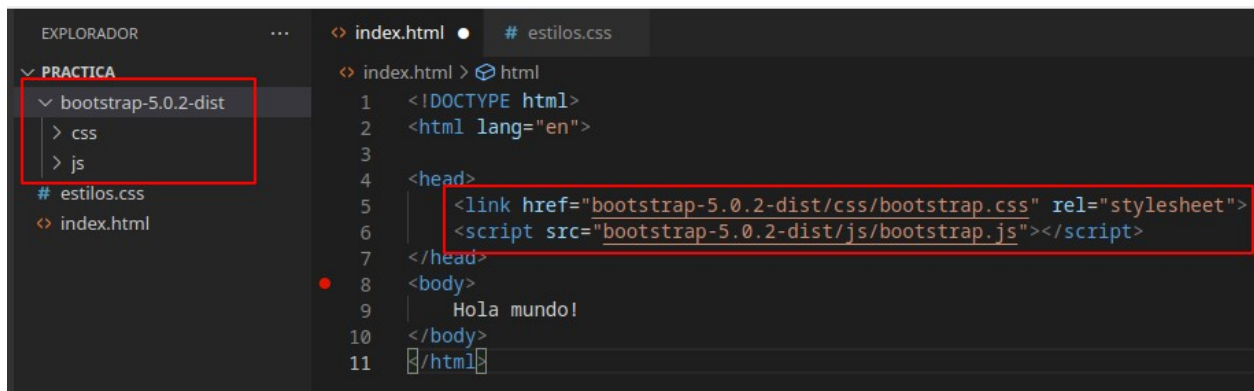
Download Bootstrap to get the compiled CSS and JavaScript, source code, or include it with your favorite package managers like npm, RubyGems, and more.

Compiled CSS and JS

Download ready-to-use compiled code for **Bootstrap v5.0.2** to easily drop into your project, which includes:

- Compiled and minified CSS bundles (see [CSS files comparison](#))
- Compiled and minified JavaScript plugins (see [JS files comparison](#))

This doesn't include documentation, source files, or any optional JavaScript dependencies like Popper.

[Download](#)

```
EXPLORADOR
...
index.html
# estilos.css

PRACTICA
  bootstrap-5.0.2-dist
    css
    js
  # estilos.css
  index.html

index.html
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <link href="bootstrap-5.0.2-dist/css/bootstrap.css" rel="stylesheet">
6   <script src="bootstrap-5.0.2-dist/js/bootstrap.js"></script>
7 </head>
8 <body>
9   Hola mundo!
10 </body>
11 </html>
```

~~¿Qué es un framework CSS?~~

~~Introducción a Bootstrap~~

Breakpoints

Contenedores

Grillas

Colores, botones y bordes



Breakpoints

Los puntos de interrupción o **breakpoints** en Bootstrap, son anchos personalizables que permiten definir como se adapta el diseño HTML al tamaño del dispositivo desde donde se accede.

- Los puntos de interrupción son los componentes básicos del diseño responsivo. Permiten controlar como se adapta el diseño a una ventana gráfica o dispositivo particular.
- Basado en **media queries**. Permiten aplicar estilos en función de un conjunto de parámetros del navegador y del sistema operativo.
- Mobile first (primero el diseño para dispositivos móviles). Utilizar el mínimo de diseño y luego agregar capas de estilos para dispositivos más grandes.

Breakpoint	Class infix	Dimensions
X-Small	<i>None</i>	<576px
Small	sm	≥576px
Medium	md	≥768px
Large	lg	≥992px
Extra large	xl	≥1200px
Extra extra large	xxl	≥1400px





~~¿Qué es un framework CSS?~~

~~Introducción a Bootstrap~~

~~Breakpoints~~

Contenedores

Grillas

Colores, botones y bordes





Contenedores

Los contenedores son una pieza de construcción fundamental de Bootstrap. Permiten contener, rellenar y alinear su contenido dentro de un dispositivo o ventana determinada.

Son el elemento de diseño más básico que existe. Y si bien los contenedores se pueden anidar (*Padre* → *Hijo (Padre)* → *Hijo (Segunda línea)*), la mayoría de los diseños no requieren un contenedor anidado.

En Bootstrap existen tres clases de contenedores:

- **.container**: establece un valor de **max-width** en cada punto de interrupción (breakpoint).
- **.container-fluid**: ocupa todo el ancho del tamaño de la pantalla (**width:100%;**).
- **.container-{breakpoint}**: ocupa todo el ancho del tamaño de pantalla, excepto del breakpoint que se especifique.

*Estas clases permiten que el elemento HTML actúe de forma responsiva, ajustando el contenido siempre al ancho del contenedor. Esto quiere decir que el valor de **max-width** cambia en cada breakpoint*





Contenedores (Continuación...)

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



Contenedores (Continuación...)

Veamos un ejemplo:

```
<div class="container-sm bg-danger">100% del ancho hasta breakpoint small (menor que 576px)</div>  
<div class="container-md bg-primary">100% del ancho hasta breakpoint medium (menor que 768px)</div>  
<div class="container-lg bg-secondary">100% del ancho hasta breakpoint large (menor que 992px)</div>  
<div class="container-xl bg-danger">100% del ancho hasta breakpoint extra large (menor que 1200px)</div>  
<div class="container-xxl bg-primary">100% del ancho hasta breakpoint extra extra large (menor que 1400px)</div>
```



El ancho de los elementos **<div>** se adapta automáticamente de acuerdo al tamaño de la pantalla del navegador o del dispositivo desde donde se acceda.

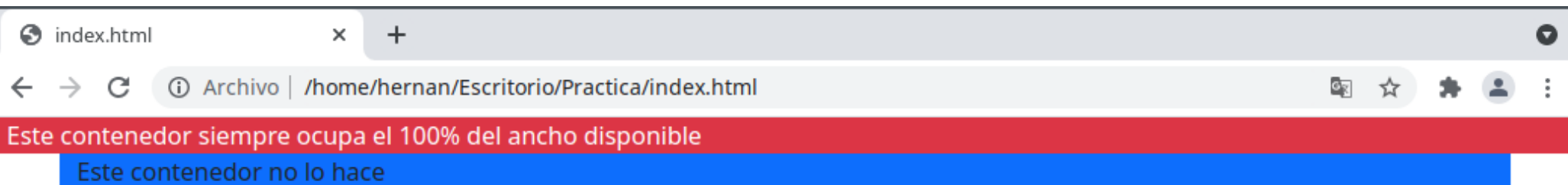




Contenedores (Continuación...)

Veamos otro ejemplo:

```
<div class="container-fluid bg-danger text-white">  
  Este contenedor siempre ocupa el 100% del ancho disponible  
</div>  
<div class="container bg-primary">  
  Este contenedor no lo hace  
</div>
```



Como vemos, **.container-fluid** utiliza siempre todo el ancho disponible, mientras que **.container** hace uso de los puntos de interrupción.





~~¿Qué es un framework CSS?~~

~~Introducción a Bootstrap~~

~~Breakpoints~~

~~Contenedores~~

Grillas

Colores, botones y bordes





Grillas

El sistema de cuadrícula (**grid**) de Bootstrap utiliza contenedores, filas y columnas para diseñar y alinear el contenido. Está construido con **flexbox** y es totalmente responsivo.

Veamos el siguiente ejemplo:

Contenedor principal

Contenedor columna

```
<div class="container">
  <div class="row">
    <div class="col">Column</div>
    <div class="col">Column</div>
    <div class="col">Column</div>
  </div>
</div>
```

Contenedor fila





Grillas (Continuación...)

Del ejemplo anterior vemos que se han creado tres columnas de igual ancho para todos los dispositivos y ventanas gráficas, utilizando las clases de cuadrícula.

- **.col** → define un contenedor como una columna de la cuadrícula
- **.row** → define un contenedor como una fila de la cuadrícula



*Las columnas están centradas y se adaptan al tamaño de la pantalla gracias al contenedor principal **.container***

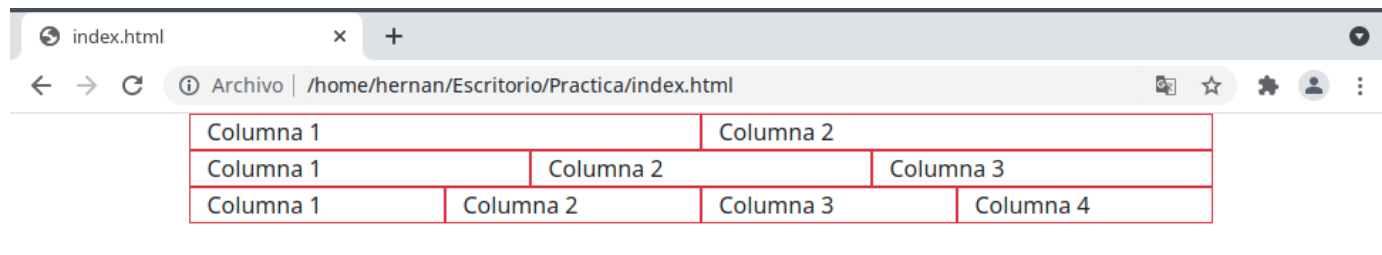


Grillas (Continuación...)

¿Como funciona?

- Los contenedores centran y rellenan horizontalmente su contenido. Utilizando la clase **.container** podemos definir un ancho responsivo, **.container-fluid** para utilizar el 100% del ancho disponible o bien **.container-{breakpoint}** para adaptar el diseño responsivo de manera específica.
- Las filas envuelven a las columnas. Cada columna presenta un relleno (padding) horizontal para controlar el espacio entre ellas.
- Existen 12 columnas disponibles por filas, lo que permite crear diferentes combinaciones de elementos que abarquen cualquier número de columnas.

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



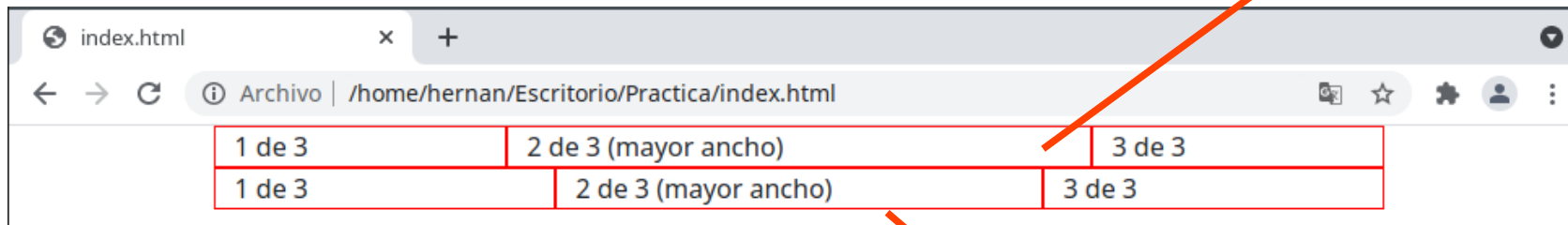


Grillas (Continuación...)

También podemos definir un ancho específico a cualquier de nuestras columnas con la clase **col-x** (donde *x* es *un valor entre 1 y 12*). Aquellas que no tengan definido este atributo se adaptan automáticamente al espacio disponible de la fila.

```
<div class="container">
  <div class="row">
    <div class="col">1 de 3</div>
    <div class="col-6">2 de 3 (mayor ancho)</div>
    <div class="col">3 de 3</div>
  </div>
  <div class="row">
    <div class="col">1 de 3</div>
    <div class="col-5">2 de 3 (mayor ancho)</div>
    <div class="col">3 de 3</div>
  </div>
</div>
```

Tamaño 6 de 12 unidades disponibles



Tamaño 5 de 12 unidades disponibles





Grillas (Continuación...)

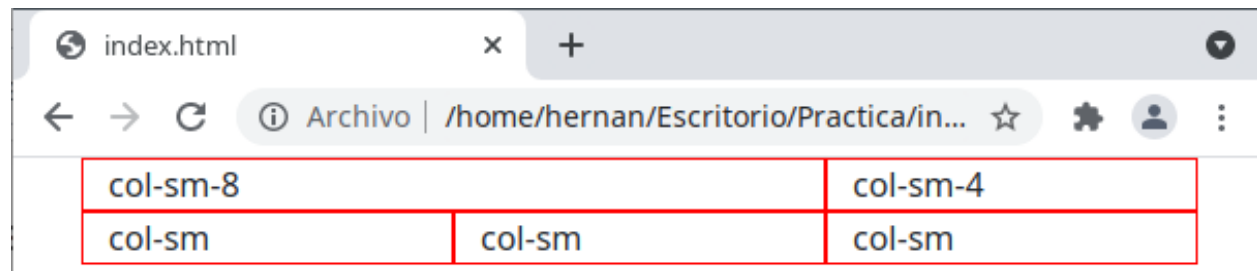
La cuadrícula de Bootstrap permite utilizar breakpoints para el diseño de las columnas, donde podremos definir el comportamiento que tendrán las mismas en relación al tamaño de pantalla o dispositivo del que se acceda.

Para esto debemos definir la clase que tendrá nuestro contenedor de columna de la siguiente manera:

col-{breakpoint}-[tamaño] → *El tamaño es opcional*

Ejemplo:

```
<div class="container">
  <div class="row">
    <div class="col-sm-8">col-sm-8</div>
    <div class="col-sm-4">col-sm-4</div>
  </div>
  <div class="row">
    <div class="col-sm">col-sm</div>
    <div class="col-sm">col-sm</div>
    <div class="col-sm">col-sm</div>
  </div>
</div>
```



La primera fila contiene dos columnas con un tamaño de 8 y 4 unidades respectivamente para resoluciones **sm** (**mayor o igual a 576px**).

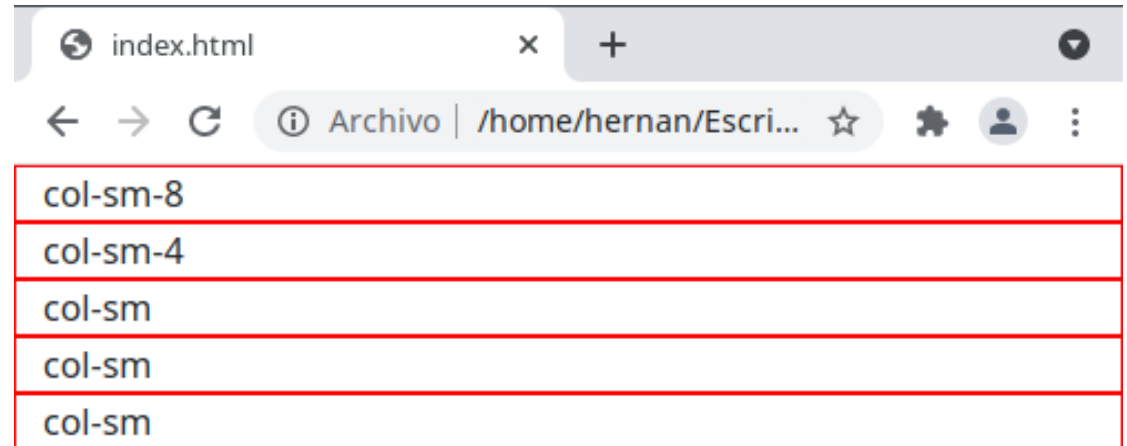
La segunda fila presenta tres columnas que se ajustan automáticamente al ancho disponible si el tamaño de pantalla es **sm** (**mayor o igual a 576px**).



Grillas (Continuación...)

Continuando con el mismo ejemplo si reducimos el tamaño de la pantalla podemos notar que cada una de las columnas ocuparan el ancho total disponible de la fila. Apilándose una sobre la otra

```
<div class="container">
  <div class="row">
    <div class="col-sm-8">col-sm-8</div>
    <div class="col-sm-4">col-sm-4</div>
  </div>
  <div class="row">
    <div class="col-sm">col-sm</div>
    <div class="col-sm">col-sm</div>
    <div class="col-sm">col-sm</div>
  </div>
</div>
```



Esto sucede debido a que el tamaño actual de la pantalla es menor al breakpoint “**.sm**” que establecimos.



~~¿Qué es un framework CSS?~~

~~Introducción a Bootstrap~~

~~Breakpoints~~

~~Contenedores~~

~~Grillas~~

Colores, botones y bordes

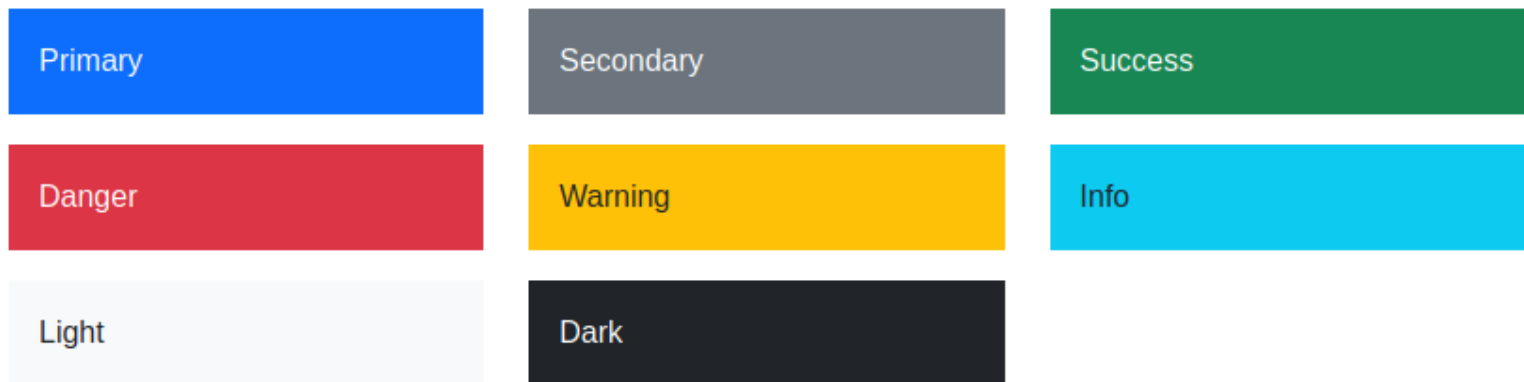




Bootstrap colors

Bootstrap posee un sistema de colores definidos en clases CSS que permiten diseñar estilos y componentes. Permite una personalización más ágil y directa.

Aunque exista una gran variedad de colores disponibles, estos son los temas principales.



Cada color está identificado por el nombre de una clase. A continuación veremos como aplicar estas clases en diversos elementos HTML.

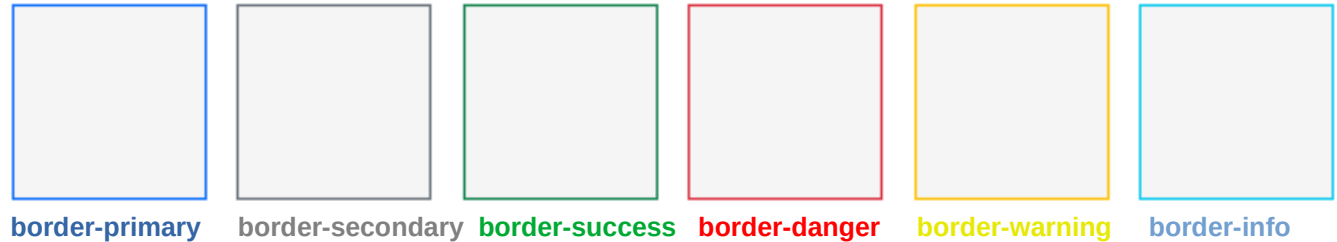




Bootstrap colors (Continuación...)

A continuación veremos como se aplican las diferentes clases de colores de Bootstrap en varios componentes HTML.

- Bordes
border-{color}



- Texto
text-{color}

text-primary text-secondary text-success text-danger text-warning text-info

- Color de fondo
bg-{color}



- Botones
btn-{color}





Bootstrap colors (Continuación...)

Veamos un ejemplo:

Index.html

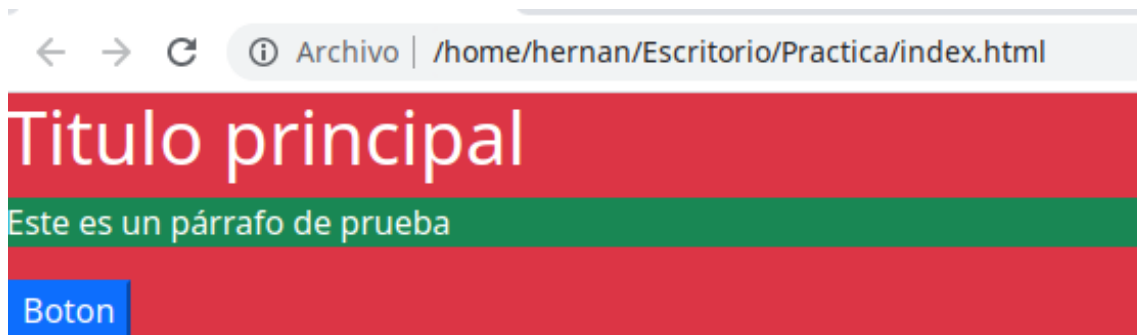
```
<!DOCTYPE html>
<html lang="en">

<head>
  <link href="bootstrap-5.0.2-dist/css/bootstrap.css" rel="stylesheet">
  <script src="bootstrap-5.0.2-dist/js/bootstrap.js"></script>
</head>
<body>

  <div class="bg-danger">
    <h1 class="text-white">Titulo principal</h1>
    <p class="bg-success text-white">Este es un párrafo de prueba</p>
    <button class="btn-primary">Boton</button>
  </div>

</body>
</html>
```

Resulta sencillo aplicar estilos sobre ciertos componentes HTML, sin la necesidad de definir una hoja de estilos CSS.

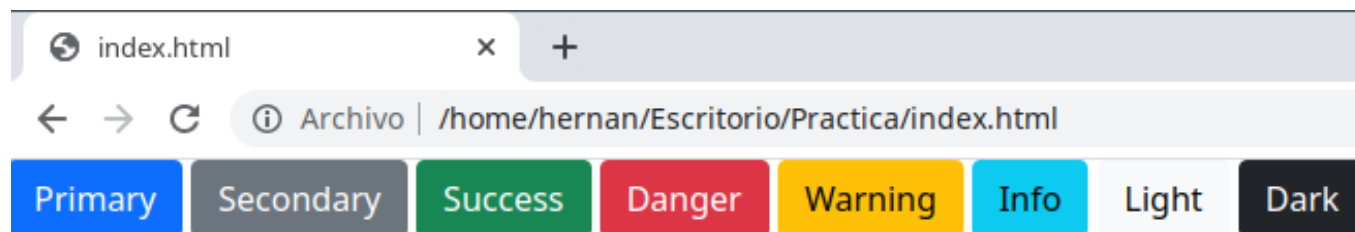




Bootstrap buttons

Bootstrap incluye una gran variedad de estilos predefinidos para los botones. Para esto debemos utilizar la clase **.btn** seguido de **.btn-{color}** sobre un elemento `<button>`.

```
<button type="button" class="btn btn-primary">Primary</button>
<button type="button" class="btn btn-secondary">Secondary</button>
<button type="button" class="btn btn-success">Success</button>
<button type="button" class="btn btn-danger">Danger</button>
<button type="button" class="btn btn-warning">Warning</button>
<button type="button" class="btn btn-info">Info</button>
<button type="button" class="btn btn-light">Light</button>
<button type="button" class="btn btn-dark">Dark</button>
```

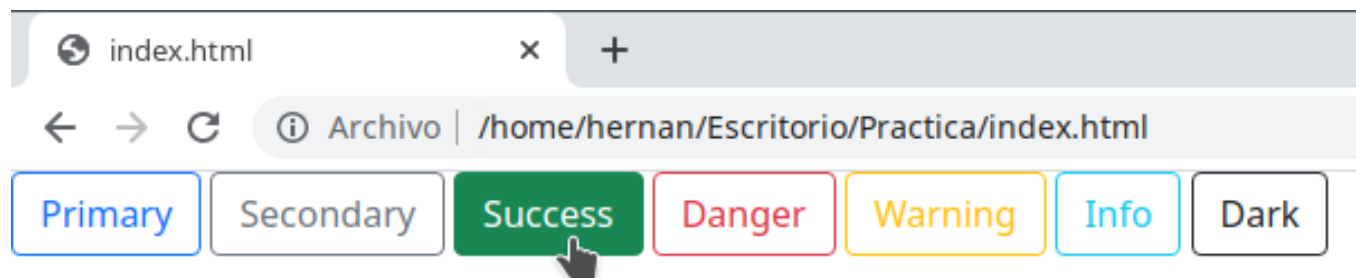




Bootstrap buttons

También contamos con clases para estilizar botones cuando posicionamos el mouse o hacemos foco sobre ellos. En este caso utilizaremos **.btn** seguido de **.btn-outline-{color}**.

```
<button type="button" class="btn btn-outline-primary">Primary</button>
<button type="button" class="btn btn-outline-secondary">Secondary</button>
<button type="button" class="btn btn-outline-success">Success</button>
<button type="button" class="btn btn-outline-danger">Danger</button>
<button type="button" class="btn btn-outline-warning">Warning</button>
<button type="button" class="btn btn-outline-info">Info</button>
<button type="button" class="btn btn-outline-light">Light</button>
<button type="button" class="btn btn-outline-dark">Dark</button>
```

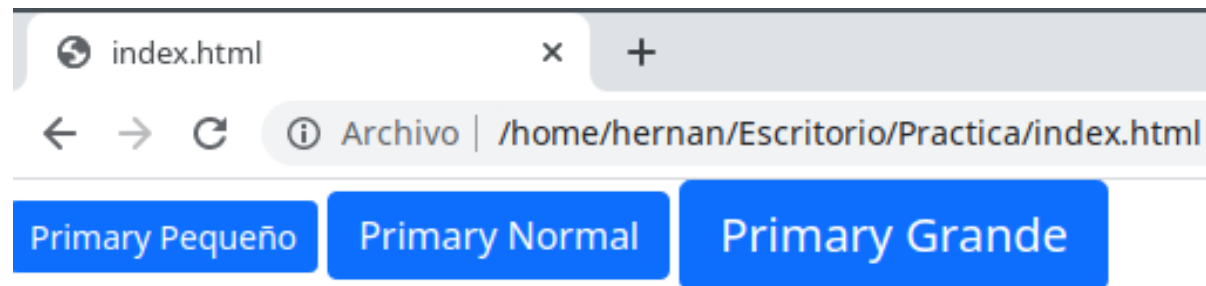




Bootstrap buttons

Los botones también cuentan con clases para modificar su tamaño predefinido. Utilizando la clase **.btn** seguido de **.btn-sm** (tamaño pequeño) ó **.btn-lg** (tamaño grande).

```
<button type="button" class="btn btn-primary btn-sm">Primary Pequeño</button>  
<button type="button" class="btn btn-primary">Primary Normal</button>  
<button type="button" class="btn btn-primary btn-lg">Primary Grande</button>
```





Bootstrap border

Bootstrap además incluye clases que permiten fácilmente diseñar los bordes de cualquier elemento HTML.

Veamos a continuación las distintas clases disponibles para el diseño de bordes:

- **.border** → agrega un borde sobre el elemento en todas las direcciones.
- **.border-top** → agrega un borde superior.
- **.border-end** → agrega un borde sobre el lateral derecho del elemento.
- **.border-bottom** → agrega un borde inferior.
- **.border-start** → agrega un borde sobre el lateral izquierdo del elemento.

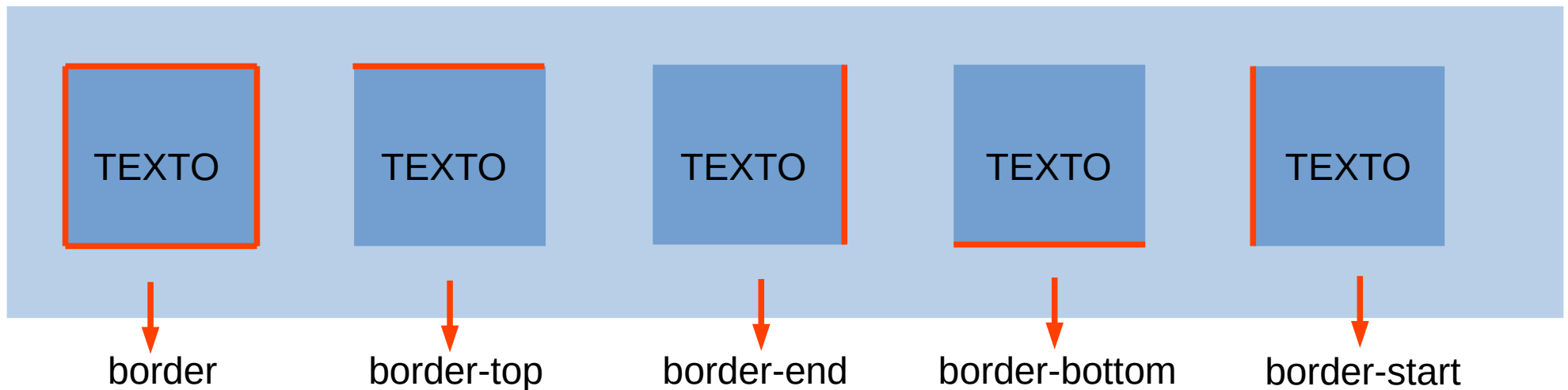




Bootstrap border

Veamos un ejemplo:

```
<span class="border">Texto</span>  
<span class="border-top">Texto</span>  
<span class="border-end">Texto</span>  
<span class="border-bottom">Texto</span>  
<span class="border-start">Texto</span>
```

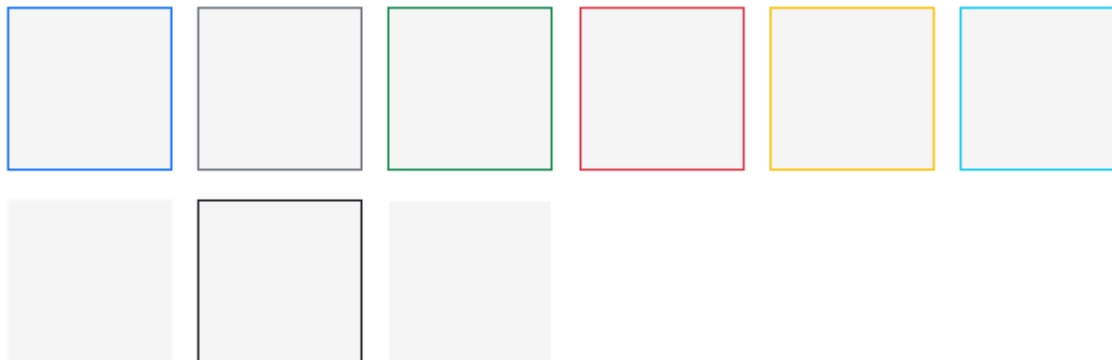




Bootstrap border

Utilizando los estilos de colores predeterminados en Bootstrap podemos agregar facilmente color a los bordes. Para este utilizaremos sobre un elemento HTML la clase **.border** seguido de **.border-{color}**.

```
<span class="border border-primary"></span>  
<span class="border border-secondary"></span>  
<span class="border border-success"></span>  
<span class="border border-danger"></span>  
<span class="border border-warning"></span>  
<span class="border border-info"></span>  
<span class="border border-light"></span>  
<span class="border border-dark"></span>  
<span class="border border-white"></span>
```



¿Consultas?

