# **­­Taller Avanzado de Bootstrap 5**

## **1. Introducción a Bootstrap 5**

### ● ¿Qué es Bootstrap?

Bootstrap es un framework de desarrollo web de código abierto que simplifica la creación de sitios responsivos y móviles. Proporciona una colección de sintaxis predefinidas para crear diseños de plantillas, lo que permite a los desarrolladores construir sitios web más rápidamente.

Con scripts en HTML, CSS y JS, Bootstrap ofrece una variedad de funciones y componentes listos para usar, liberando a los desarrolladores de la necesidad de escribir código desde cero. En resumen, Bootstrap agiliza el proceso de desarrollo al proporcionar herramientas y elementos de diseño predefinidos.

### ● Novedades en Bootstrap 5

1. **Eliminación de la compatibilidad con jQuery**: En Bootstrap 5.0, ya no se utiliza jQuery de forma nativa, lo que resulta en diseños web más rápidos y una mejora en la carga de la página.
2. **Incompatibilidad con versiones antiguas de Internet Explorer**: Se elimina el soporte para Internet Explorer 9 y 10, siendo compatible únicamente con Microsoft Edge.
3. **Introducción de variables CSS**: Bootstrap 5.0 adopta el uso de variables CSS, lo que simplifica la creación de diseños modernos sin necesidad de escribir tanto código.
4. **Mejoras en formularios**: Se añaden nuevos estilos visuales y funcionalidades a los formularios, como los interruptores ("switch"), para un aspecto más moderno.
5. **Sistema de cuadrícula mejorado**: Se añade un nuevo breakpoint ("xxl") para soportar monitores más grandes, además de modificaciones en las clases para gestionar la rejilla y estilos visuales.
6. **Cambios menores confirmados**: Incluyen mejoras en la escritura de derecha a izquierda (RTL), menús fuera de lienzo, y la posibilidad de usar código HTML en lugar de CSS para archivos SVG.

Estas son las novedades principales, pero también hay otros cambios menores que podrían ser relevantes, como el soporte mejorado para idiomas RTL y la posibilidad de utilizar HTML en lugar de CSS para archivos SVG.

### ● Configuración del entorno de trabajo

### *Instalación de Bootstrap*

Para empezar a usar Bootstrap únicamente tienes que relacionar sus scripts y hojas de estilo en la cabecera del código HTML de tu web. Existen dos formas de hacerlo: desde una CDN (Content Delivery Network) o descargando los archivos de la página oficial y añadiéndolos al hosting de tu sitio.

* *Insertar mediante CDN*

Si por cualquier motivo no quieres descargar los archivos y subirlos a tu hosting, puede incluirlos desde una CDN. Únicamente tienes que añadir el siguiente código:

<!-- Latest compiled and minified CSS -->

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0

.0/css/bootstrap.min.css">

<!-- jQuery library -->

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min

.js"></script>

<!-- Popper JS -->

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/p

opper.min.js"></script>

<!-- Latest compiled JavaScript -->

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.

min.js"></script>

Recuerda que también debes incluir la dependencia a jQuery, ya que el javascript está escrito en ese lenguaje.

Una de las ventajas de emplear una CDN es que la propia red se preocupa de administrarte los archivos que se encuentren más cerca para reducir los posibles tiempos de carga. Si tienes tu web alojada en dinahosting despreocúpate, pues poseemos unas tasas de transferencia muy rápidas, por lo que esto no será un problema para ti.

* *Descargando los archivos Bootstrap*

Si quieres descargar y alojar Bootstrap es tan sencillo como descargar los archivos del repositorio en su página web, y después añadir la ruta donde los tienes alojados en tu HTML. En la web del framework también te indican cómo instalarlos desde consola.

Desde mi punto de vista esta es la mejor opción, ya que no dependerás de terceros para proporcionar los archivos necesarios que necesita tu web para su correcta visualización.

## **2. Sistema de Grid Avanzado**

## Semántica | HTML5

### ● [Repaso del sistema de Grid](https://dinahosting.com/blog/bootstrap-que-es-como-instalarlo-y-ejemplos-de-uso/#:~:text=Para%20empezar%20a%20usar%20Bootstrap,al%20hosting%20de%20tu%20sitio.)

*- Sistema de Grid:* El sistema de cuadrícula de Bootstrap es una herramienta esencial para crear diseños responsivos y flexibles en los sitios web. Se basa en flexbox y ofrece una serie de clases predefinidas para organizar el contenido en filas y columnas.

* Contenedor: El contenedor es el elemento principal que envuelve todo el contenido de la página. Puede ser de tipo .container para un ancho fijo o .container-fluid para ocupar todo el ancho disponible.
* Filas: Las filas son contenedores horizontales que agrupan columnas. Se definen con la clase .row y están diseñadas para contener columnas.
* Columnas: Las columnas son los elementos que contienen el contenido real. Se organizan horizontalmente dentro de las filas y están definidas por clases como .col. Bootstrap proporciona clases específicas para diferentes tamaños de dispositivos, como .col-xs-, .col-sm-, .col-md-, .col-lg-, .col-xl-, y .col-xxl-.

**.col-xs**– : dispositivos extra pequeños – ancho de pantalla inferior a 576 px

**.col-sm-** :dispositivos pequeños – ancho de pantalla igual o superior a 576 px

**.col-md-** : dispositivos medianos – ancho de pantalla igual o mayor que 768 px

**.col-lg-** : dispositivos grandes – ancho de pantalla igual o mayor que 992 px

**.col-xl-** : dispositivos grandes – ancho de pantalla igual o mayor que 1200 px

**.col-xxl-** : dispositivos grandes – ancho de pantalla igual o mayor que 1400 px

<!-- Controla el ancho de las columnas, y cómo deben mostrarse en diferentes dispositivos -->

<div class="row">

<div class="col-sm-6 col-xs-12"></div>

<div class="col-sm-6 col-xs-12"></div>

</div>

● Técnicas avanzadas de layout(disposición) con Grid:

El sistema de cuadrícula de Bootstrap también permite técnicas avanzadas para el diseño de páginas más complejas y personalizadas.

* [*Grillas anidadas*](https://www.tutorialesprogramacionya.com/bootstrap5ya/detalleconcepto.php?punto=11&codigo=11&inicio=0)*:* Es posible anidar filas y columnas dentro de otras filas y columnas para crear diseños más complejos y flexibles. Esto se logra al colocar una nueva fila dentro de una columna existente.

<div class="container">

<div class="row">

<div class="col-lg-6" style="background-color:#aaa">

<h1>Columna 1</h1>

<p>Esto es una prueba de bootstrap 5.</p>

</div>

<div class="col-lg-6" style="background-color:#bbb">

<h1>Columna 2</h1>

<p>Esto es una prueba de bootstrap 5.</p>

<div class="row">

<div class="col-lg-6" style="background-color:#ccc">

<h2>Columna 2a</h2>

<p>Esto es una prueba de bootstrap 5.</p>

</div>

<div class="col-lg-6" style="background-color:#ddd">

<h2>Columna 2b</h2>

<p>Esto es una prueba de bootstrap 5.</p>

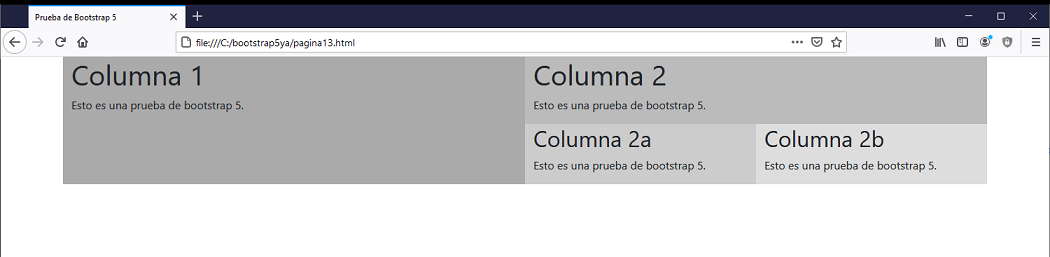
</div>

</div>

</div>

</div>

</div>



* *Colocación personalizada de columnas:* Además de utilizar las clases predefinidas para el tamaño de las columnas, puedes establecer tamaños de columna personalizados utilizando clases como .col-{breakpoint}-{size}. Esto te permite controlar el tamaño de las columnas en diferentes puntos de interrupción de manera más precisa.

En Bootstrap, puedes personalizar la colocación de las columnas utilizando las clases del sistema de rejilla (grid system). Aquí tienes una lista de las etiquetas y clases más comunes que puedes usar para diseñar un layout personalizado:

### Contenedor

* **.container**: Fijo, con anchos específicos en diferentes tamaños de pantalla.
* **.container-fluid**: Ancho completo a través de todos los tamaños de pantalla.

### Fila

* **.row**: Contenedor de columnas, asegurando que las columnas se alineen correctamente.

### Columna

Las columnas se definen utilizando clases basadas en el sistema de rejilla de Bootstrap. Puedes especificar el tamaño de las columnas y cómo se comportan en diferentes tamaños de pantalla.

* **.col**: Toma el mismo ancho en todos los tamaños de pantalla.
* **.col-\***: Columnas automáticas que ocupan igual espacio.
* **.col-[número]**: Especifica el número de columnas que ocupa (de 1 a 12).
* **.col-[tamaño]-[número]**: Especifica el tamaño de la columna en diferentes tamaños de pantalla. Los tamaños son:
  + **sm**: >= 576px
  + **md**: >= 768px
  + **lg**: >= 992px
  + **xl**: >= 1200px
  + **xxl**: >= 1400px

Ejemplo de clases de columnas:

* **.col-sm-6**: Ocupa 6 columnas en tamaños de pantalla pequeños y más grandes.
* **.col-md-4**: Ocupa 4 columnas en tamaños de pantalla medianos y más grandes.
* **.col-lg-3**: Ocupa 3 columnas en tamaños de pantalla grandes y más grandes.

### Clases de Orden

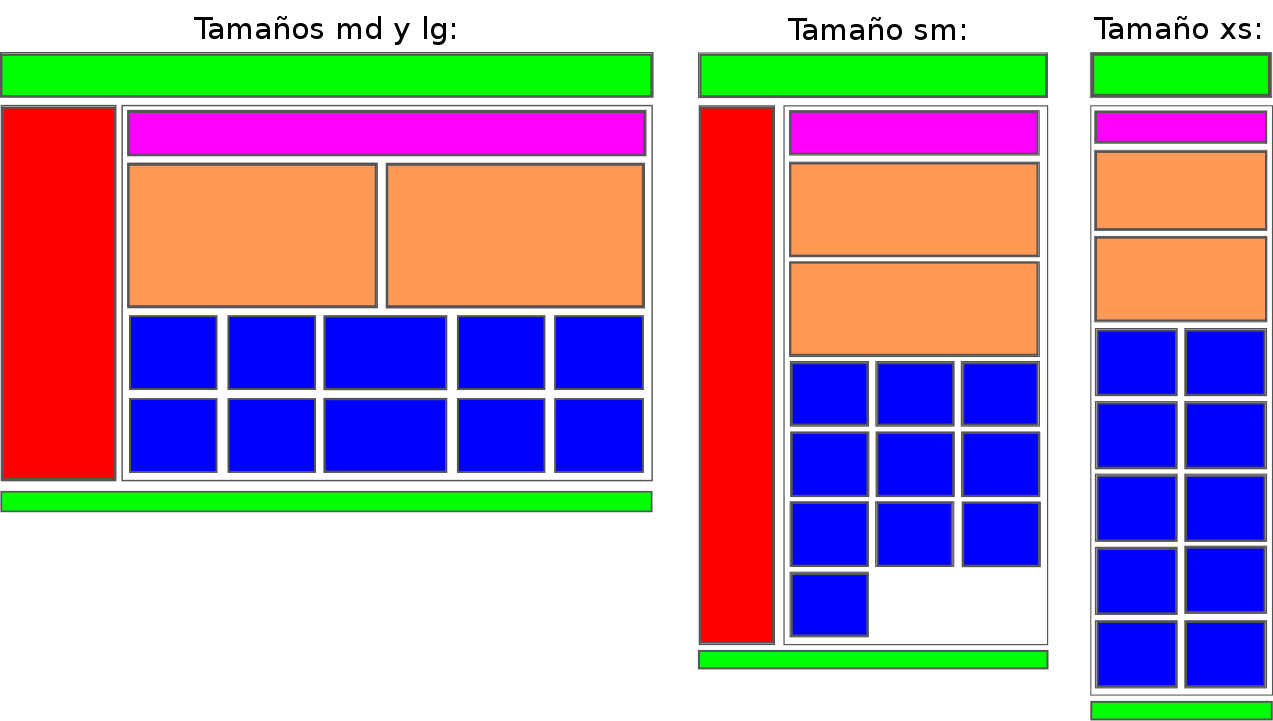
* **.order-[número]**: Cambia el orden de las columnas (de 0 a 12).
* **.order-first**: Mueve la columna al inicio.
* **.order-last**: Mueve la columna al final.
* **.order-[tamaño]-[número]**: Cambia el orden de las columnas en diferentes tamaños de pantalla.

### Clases de Alineación y Espaciado

* **.align-items-start**: Alinea los elementos al inicio de la fila.
* **.align-items-center**: Alinea los elementos al centro de la fila.
* **.align-items-end**: Alinea los elementos al final de la fila.
* **.justify-content-start**: Justifica el contenido al inicio de la fila.
* **.justify-content-center**: Justifica el contenido al centro de la fila.
* **.justify-content-end**: Justifica el contenido al final de la fila.
* **.justify-content-around**: Espacia uniformemente con espacio alrededor.
* **.justify-content-between**: Espacia uniformemente sin espacio alrededor.
* **.justify-content-evenly**: Espacia uniformemente con igual espacio entre elementos.

### Clases de Espaciado

* **.m-[número]**: Margen en todos los lados.
* **.mt-[número]**: Margen superior.
* **.mb-[número]**: Margen inferior.
* **.ml-[número]**: Margen izquierdo.
* **.mr-[número]**: Margen derecho.
* **.mx-[número]**: Margen horizontal.
* **.my-[número]**: Margen vertical.
* **.p-[número]**: Padding en todos los lados.
* **.pt-[número]**: Padding superior.
* **.pb-[número]**: Padding inferior.
* **.pl-[número]**: Padding izquierdo.
* **.pr-[número]**: Padding derecho.
* **.px-[número]**: Padding horizontal.
* **.py-[número]**: Padding vertical.



* *Offset, orden y alineación:* Bootstrap proporciona clases para el desplazamiento de columnas (offset), el orden de las columnas (order) y la alineación de contenido dentro de las columnas (align-self). Estas clases te permiten ajustar el diseño y la apariencia de tu página con mayor detalle.
* [**Offset:**](https://platzi.com/tutoriales/1331-bootstrap/11762-bootstrap-v51-layout-offset-y-margin-utilities/)Esto significa compensación. Los offset nos permiten colocar un espacio de izquierda a derecha a las columnas.

El sistema de rejilla de Bootstrap permite "desplazar" (offset) columnas hacia la derecha utilizando las clases offset. Esto es útil cuando deseas dejar un espacio vacío a la izquierda de una columna.

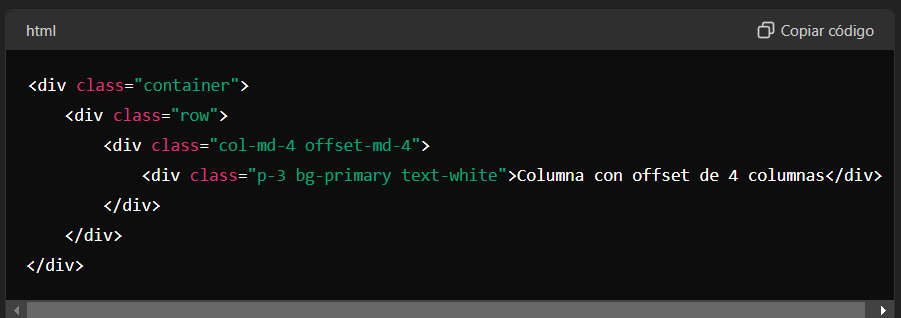
*Clases de Offset:*

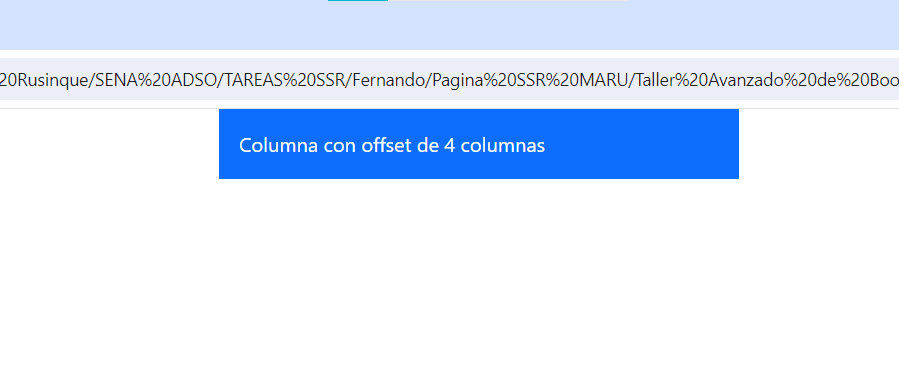
.offset-[tamaño]-[número]: Desplaza la columna el número especificado de columnas en el tamaño de pantalla especificado.

Ejemplos:

.offset-sm-2: Desplaza la columna 2 columnas hacia la derecha en tamaños de pantalla pequeños y más grandes.

.offset-md-4: Desplaza la columna 4 columnas hacia la derecha en tamaños de pantalla medianos y más grandes.





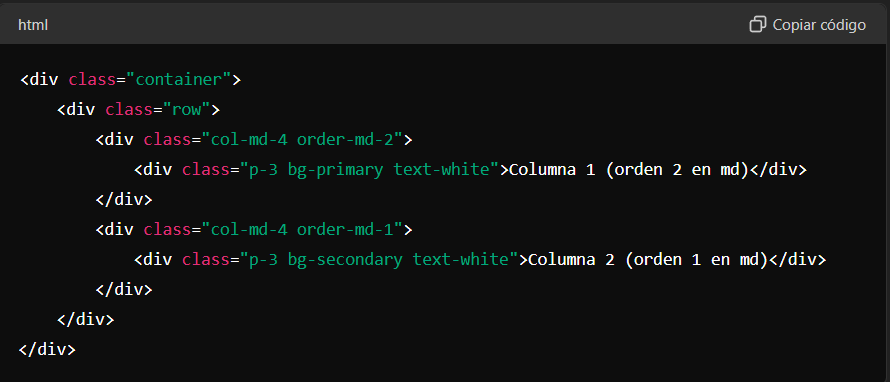
* **Orden:** Puedes cambiar el orden de las columnas utilizando las clases de orden. Esto es útil cuando deseas que las columnas se muestren en un orden diferente al especificado en el HTML, especialmente en diferentes tamaños de pantalla.

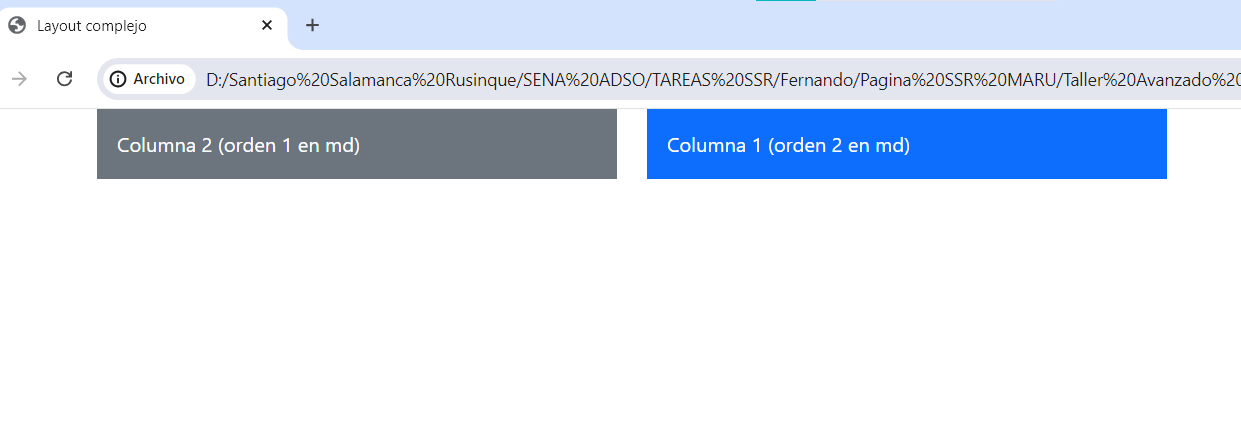
**Clases de Orden:**

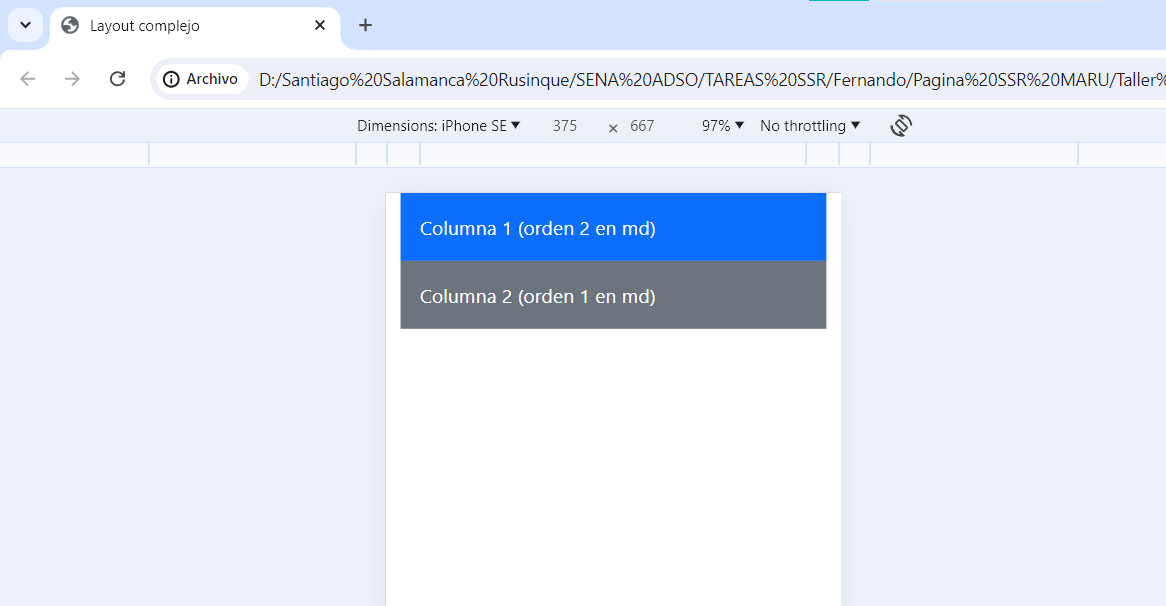
* **.order-[número]**: Cambia el orden de la columna (de 0 a 12).
* **.order-first**: Mueve la columna al inicio.
* **.order-last**: Mueve la columna al final.
* **.order-[tamaño]-[número]**: Cambia el orden de la columna en diferentes tamaños de pantalla.

Ejemplos:

* **.order-1**: Ordena la columna en la posición 1.
* **.order-md-2**: Ordena la columna en la posición 2 en tamaños de pantalla medianos y más grandes.







* **Alineación**: Bootstrap proporciona clases de alineación para alinear los elementos vertical y horizontalmente dentro de las filas y columnas.

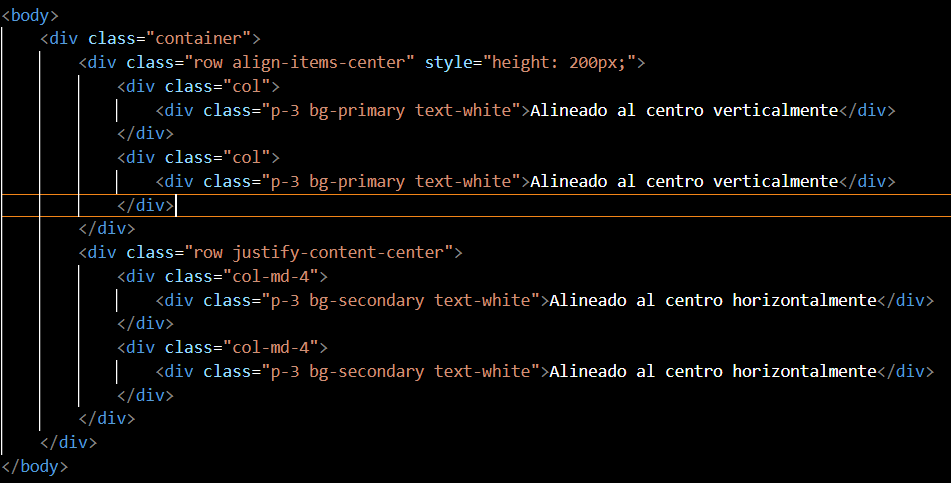
**Alineación Vertical (align-items):**

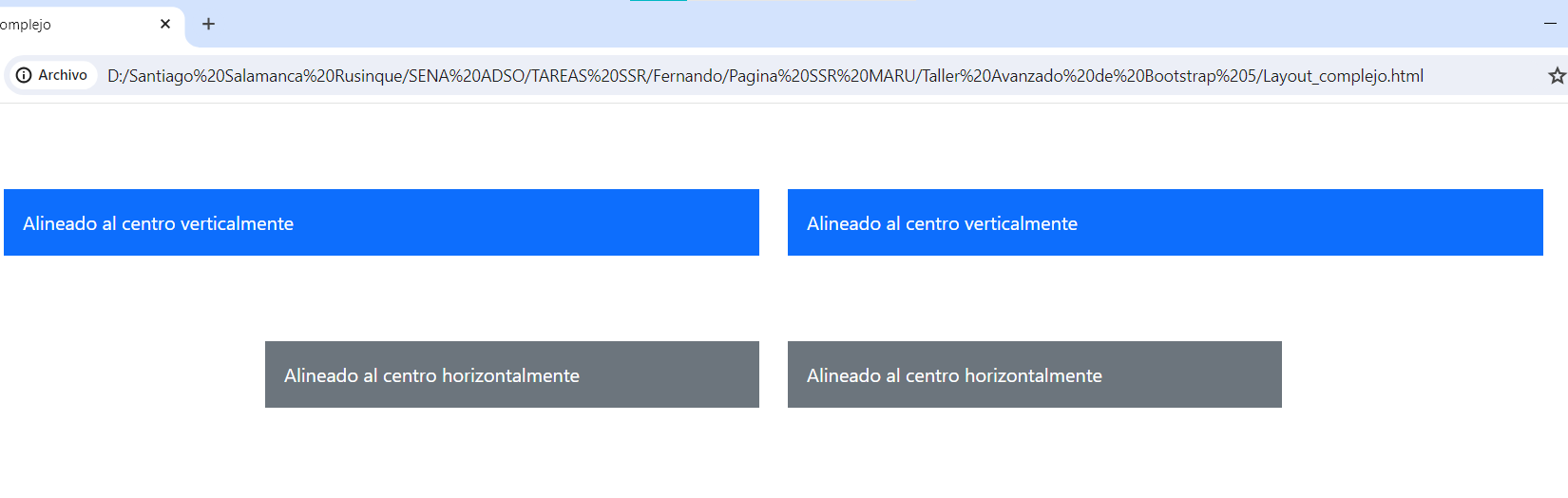
* **.align-items-start**: Alinea los elementos al inicio de la fila.
* **.align-items-center**: Alinea los elementos al centro de la fila.
* **.align-items-end**: Alinea los elementos al final de la fila.

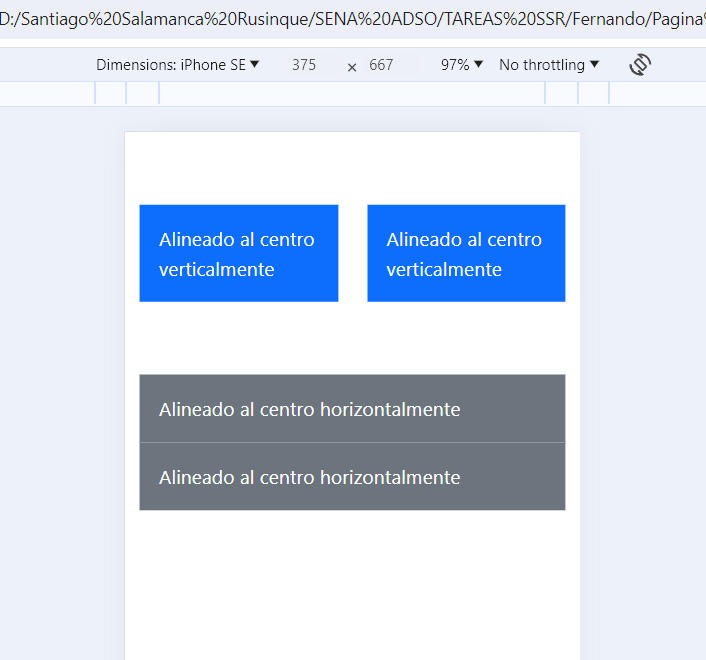
**Alineación Horizontal (justify-content):**

* **.justify-content-start**: Justifica el contenido al inicio de la fila.
* **.justify-content-center**: Justifica el contenido al centro de la fila.
* **.justify-content-end**: Justifica el contenido al final de la fila.
* **.justify-content-around**: Espacia uniformemente con espacio alrededor.
* **.justify-content-between**: Espacia uniformemente sin espacio alrededor.
* **.justify-content-evenly**: Espacia uniformemente con igual espacio entre elementos.

Ejemplo de alineación vertical y horizontal:







● Técnicas avanzadas de layout(disposición) con Grid:

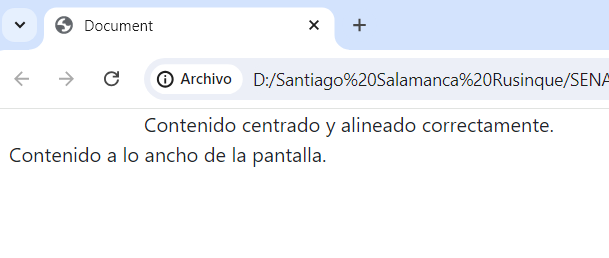
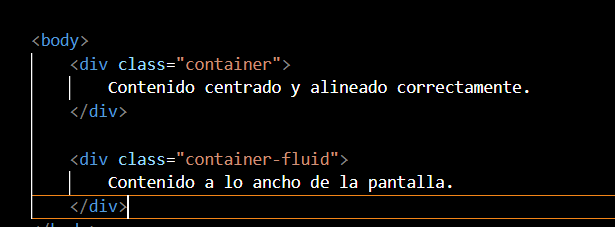
## **3. Componentes de Bootstrap**

### ● Repaso de componentes básicos

Bootstrap 5 ofrece una amplia gama de componentes básicos que puedes utilizar para crear interfaces web atractivas y funcionales. Aquí tienes una lista de algunos de los componentes más importantes y comunes que Bootstrap 5 proporciona, junto con ejemplos de uso.

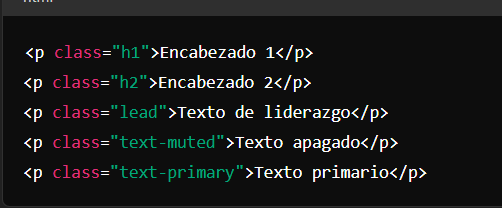
**1. Contenedores (Containers)**

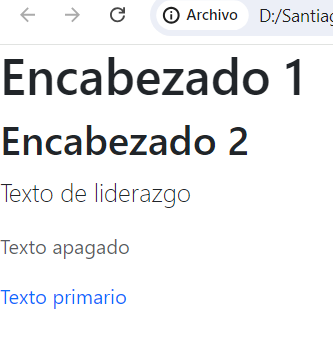
Los contenedores son elementos básicos en Bootstrap que permiten centrar y alinear correctamente tu contenido.



**2. Tipografía**

Bootstrap proporciona clases para estilizar el texto.





**3. Imágenes**

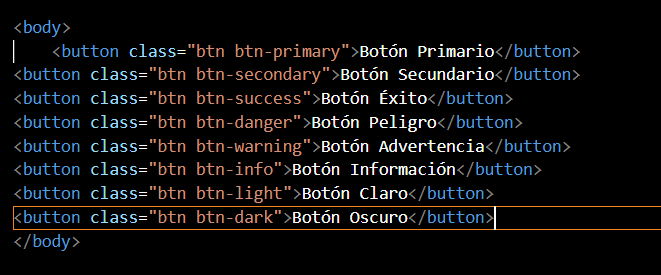
Clases para hacer que las imágenes sean responsivas y redondeadas.

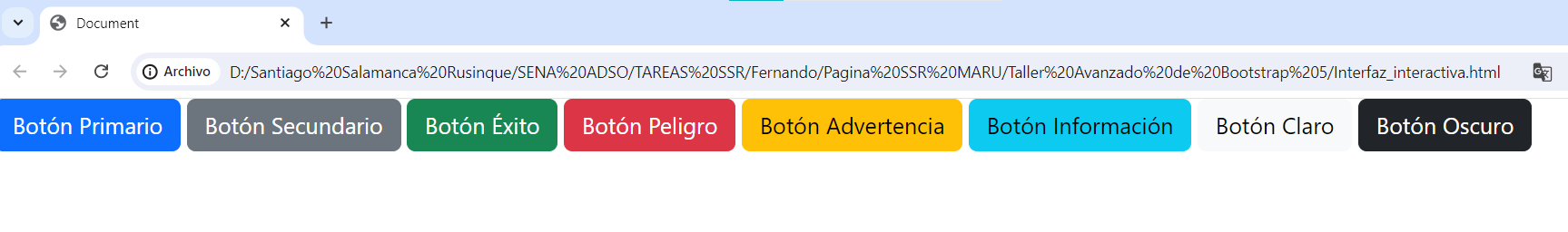




**4. Botones**

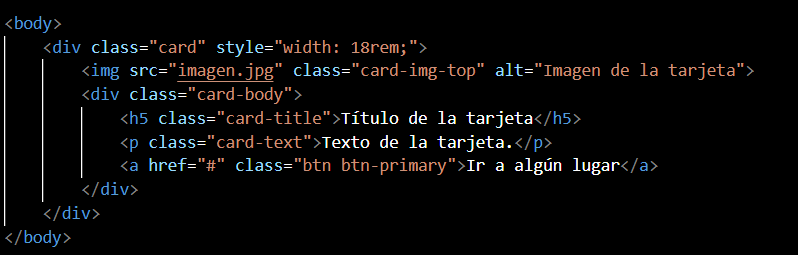
Bootstrap incluye una variedad de clases para estilizar botones.

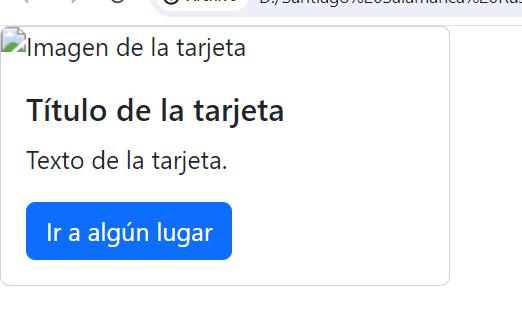




**5. Tarjetas (Cards)**

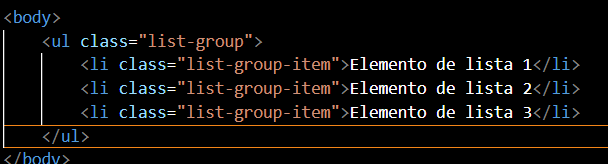
Las tarjetas son un componente flexible y extensible que contiene cualquier combinación de encabezados, texto, imágenes, listas de grupos y pies de página.

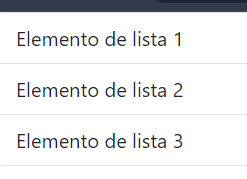


****

**6. Listas de Grupos (List Groups)**

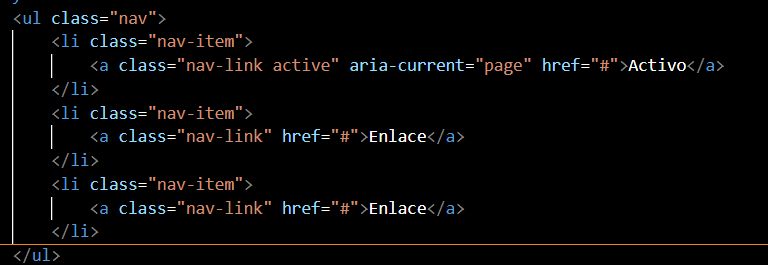
Listas de grupos son útiles para mostrar una serie de contenidos relacionados.

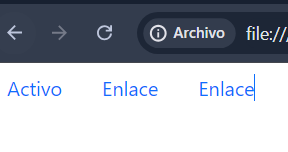
****

****

**7. Navegación (Navs)**

Bootstrap proporciona componentes para crear barras de navegación.

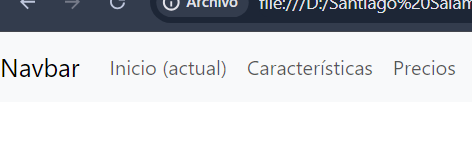


****

**8. Barra de Navegación (Navbar)**

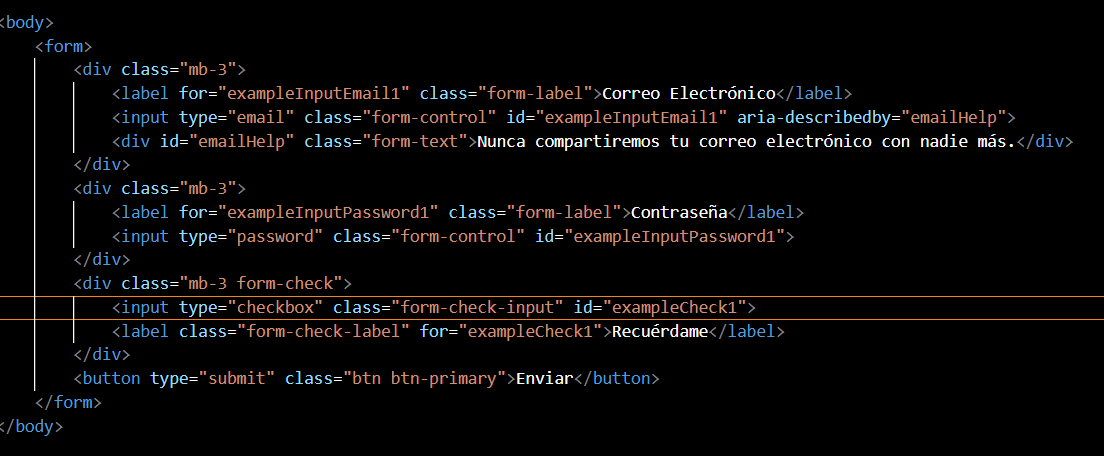
Componente de barra de navegación completo con soporte responsivo.

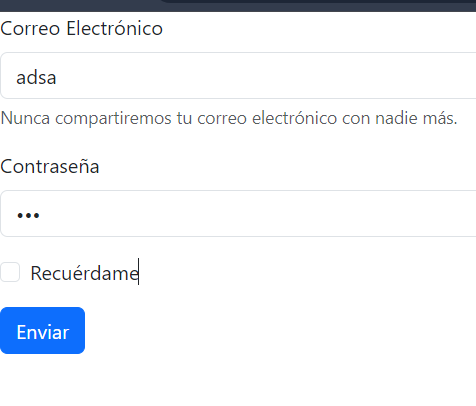
****

****

**9. Formularios**

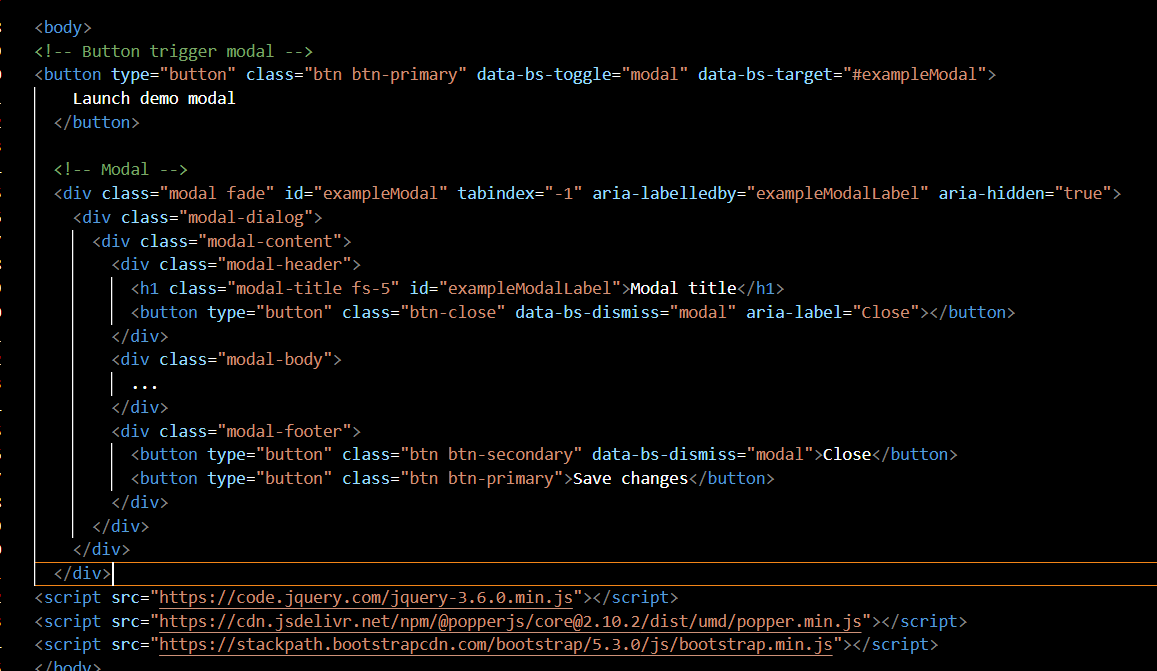
Bootstrap proporciona una variedad de clases y componentes para construir formularios.

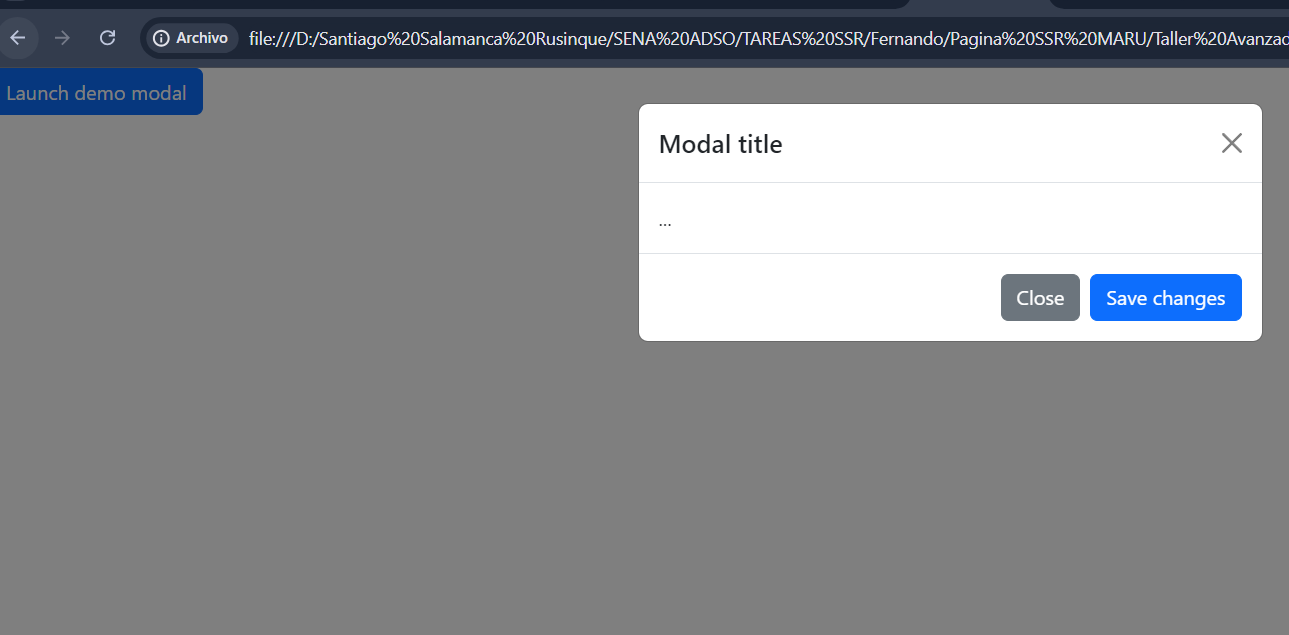


****

**10. Modales (Modals)**

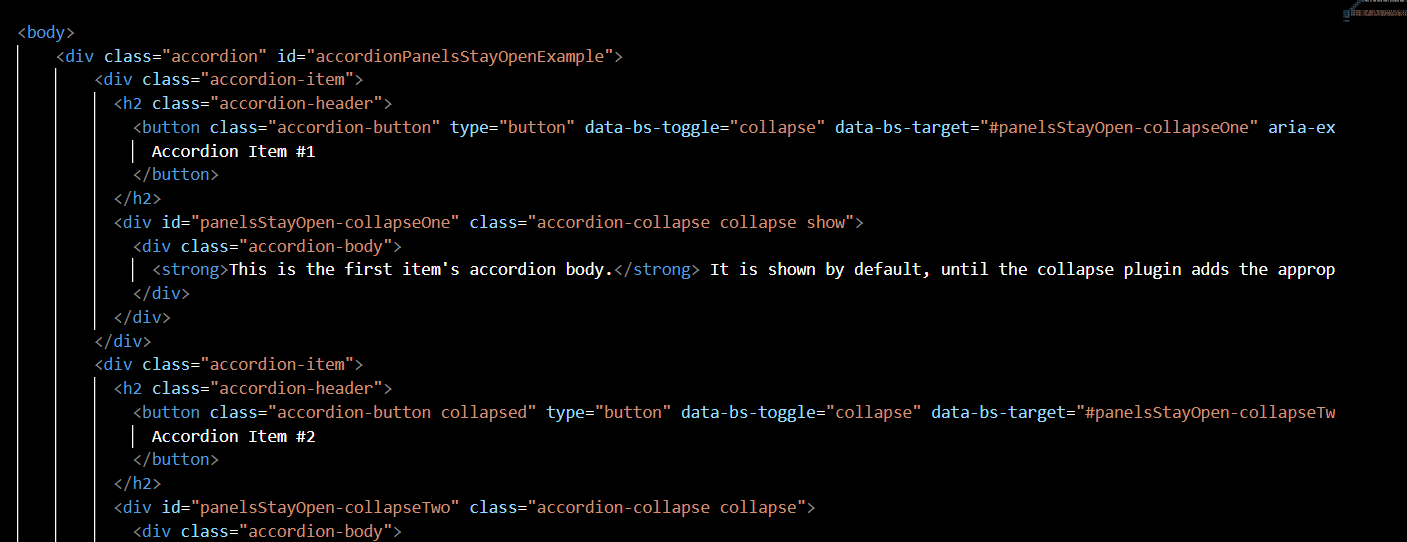
Bootstrap incluye soporte para modales, diálogos con estilo y ventanas emergentes.

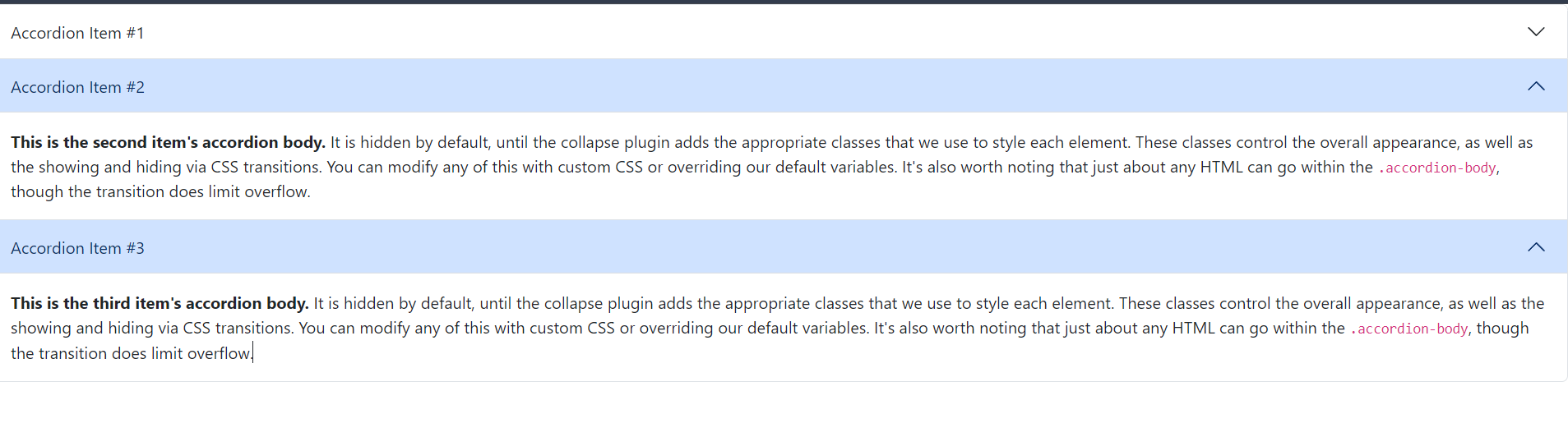




**11.** [**Acordeón (Accordion)**](https://getbootstrap.com/docs/5.3/components/accordion/)

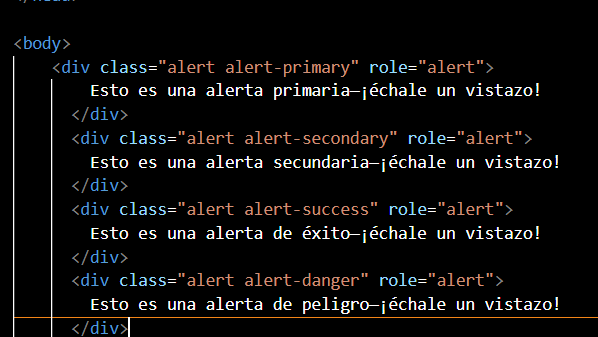
Para crear un conjunto de elementos colapsables.

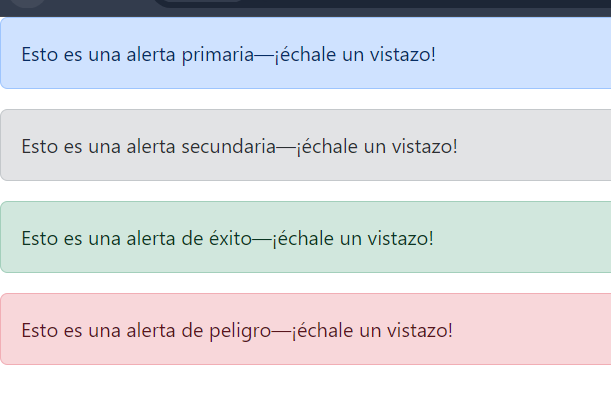




**12. Alertas (Alerts)**

Para mostrar mensajes de alerta.



****

### ● Uso avanzado de componentes:

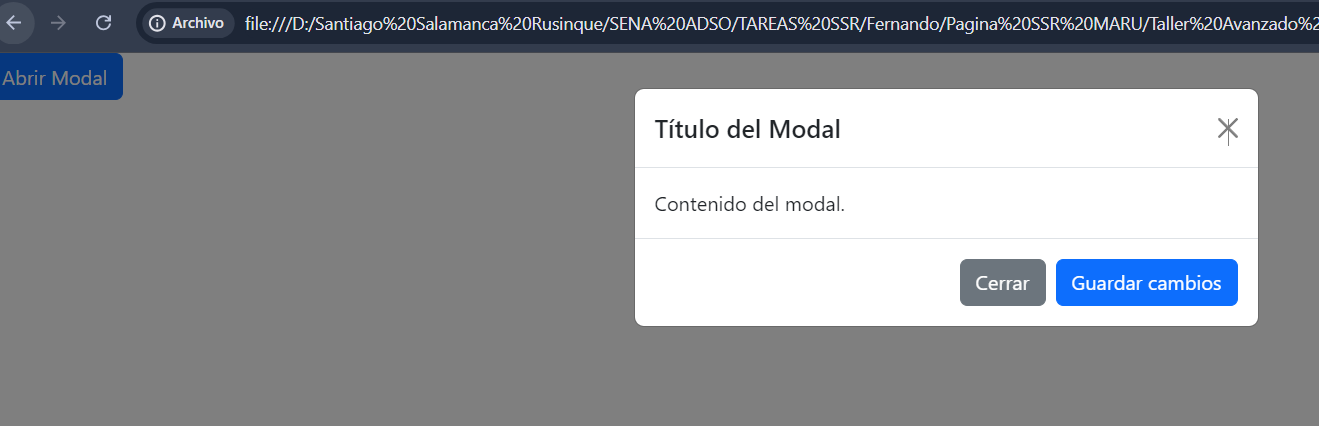
* **Modals:** Los modales son cuadros de diálogo o ventanas emergentes que se superponen sobre el contenido principal del sitio web. Bootstrap 5 proporciona un componente de modal muy versátil y fácil de usar.

#### Estructura Básica del Modal

Un modal de Bootstrap se compone de varias partes:

1. **Botón para abrir el modal**: Este botón activa el modal.
2. **Estructura del modal**: Incluye el contenedor del modal, el diálogo, el contenido del modal, la cabecera, el cuerpo y el pie del modal.

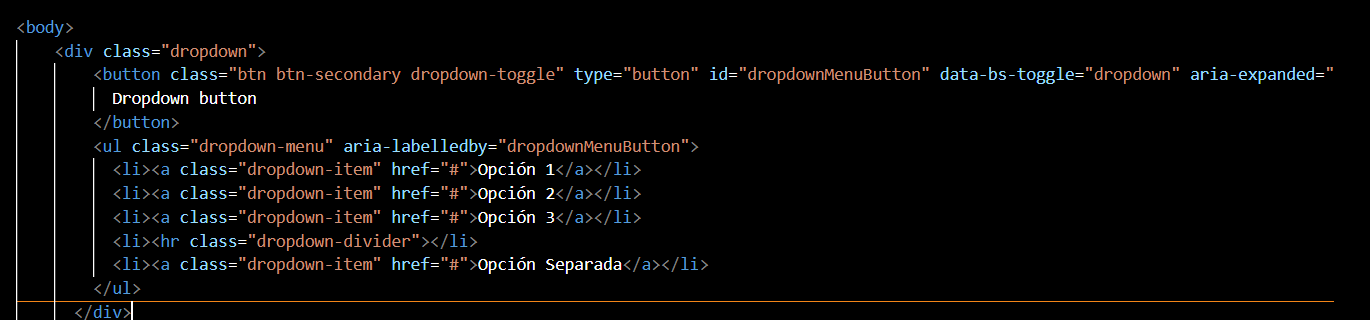


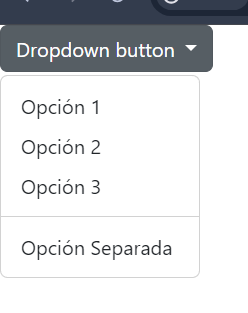


#### Componentes del Modal

* **<div class="modal fade" id="exampleModal" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">**: Este es el contenedor principal del modal. La clase modal fade hace que el modal se desvanezca cuando se muestra u oculta.
* **<div class="modal-dialog">**: Este contenedor envuelve el contenido del modal y puede tener clases adicionales para modificar su tamaño (por ejemplo, modal-lg para un modal grande, modal-sm para un modal pequeño).
* **<div class="modal-content">**: Este es el contenedor del contenido del modal.
* **<div class="modal-header">**: Esta sección contiene el encabezado del modal, incluyendo el título y el botón de cierre.
* **<div class="modal-body">**: Esta sección contiene el contenido principal del modal.
* **<div class="modal-footer">**: Esta sección contiene los botones de acción del modal.
* **Dropdowns:** Un dropdown en Bootstrap consta de varias partes:

1. **Botón de activación**: Este botón muestra el texto predeterminado o la opción seleccionada del dropdown.
2. **Lista de opciones**: Es la lista desplegable que aparece cuando se hace clic en el botón de activación.

****

****

#### **Componentes del Dropdown**

* **<div class="dropdown">**: Contenedor principal del dropdown.
* **<button class="btn btn-secondary dropdown-toggle" ...>**: Botón que activa el dropdown. La clase dropdown-toggle y los atributos data-bs-toggle y aria-expanded son esenciales para el funcionamiento del dropdown.
* **<ul class="dropdown-menu" ...>**: Lista desplegable que contiene las opciones del dropdown.
* **<li><a class="dropdown-item" href="#">Opción 1</a></li>**: Elementos de lista dentro del dropdown. Cada opción está representada por un elemento <li> con la clase dropdown-item.

#### **Dropdowns con Divisiones y Separadores**

Los dropdowns de Bootstrap permiten agregar divisiones y separadores para organizar mejor las opciones:

* **División** (<hr class="dropdown-divider">): Divide visualmente las opciones del dropdown.
* **Separador** (<li><a class="dropdown-item" href="#">Opción Separada</a></li>): Crea una opción que no es seleccionable, útil para separar grupos de opciones.

#### **Dropdowns con Alcance**

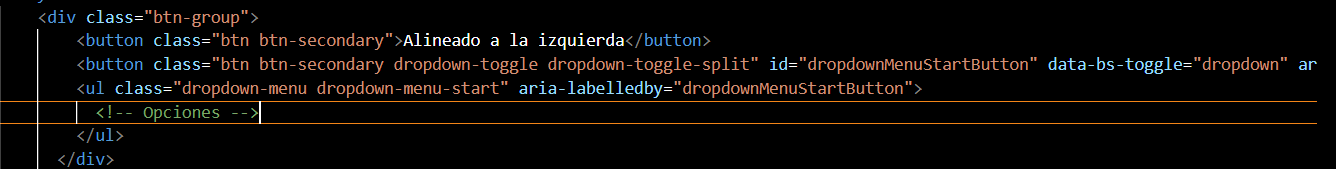
Los dropdowns pueden tener diferentes alcances y usos, dependiendo de la situación. Por ejemplo:

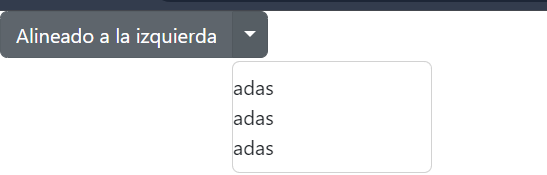
* **Dropdowns simples**: Usados para mostrar opciones y seleccionar una.
* **Dropdowns divididos**: Tienen dos partes, una que actúa como botón y otra como menú.
* **Dropdowns con alineamiento**: Pueden alinearse a la izquierda o derecha del botón.

#### **Opciones Avanzadas**

##### Dropdowns Alineados

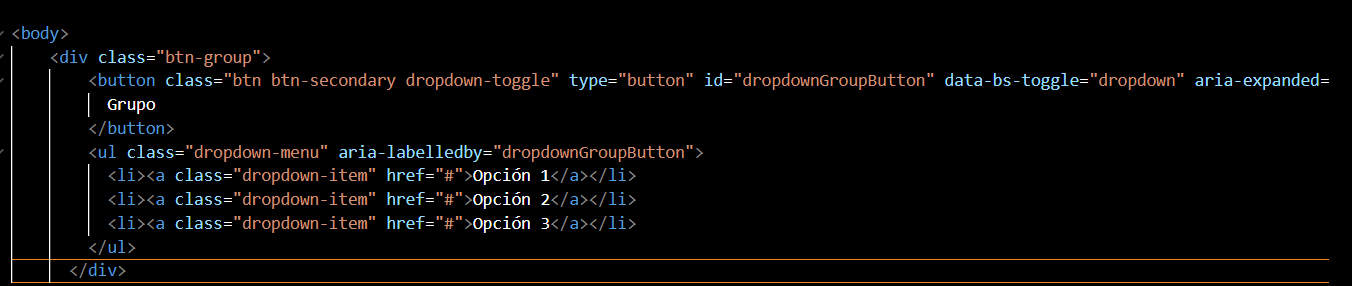
Puedes alinear los dropdowns utilizando las clases dropdown-menu-start, dropdown-menu-end, dropdown-menu-sm-start, dropdown-menu-md-start, etc.

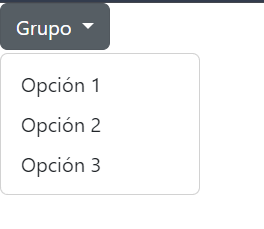




**Dropdowns con Grupos**

Puedes agrupar dropdowns en un btn-group para facilitar la administración de botones relacionados.

****

****

* **Navs y Tabs:**

¿Qué son los Navs y Tabs?

**Navs (navegaciones):** Son componentes que permiten la navegación entre diferentes secciones de contenido. Pueden estar dispuestos horizontal o verticalmente.

**Tabs (pestañas):** Son subcomponentes de los Navs que organizan el contenido en diferentes secciones dentro de una misma página. Cada pestaña muestra un contenido específico al hacer clic en ella.

Estructura Básica de Navs y Tabs

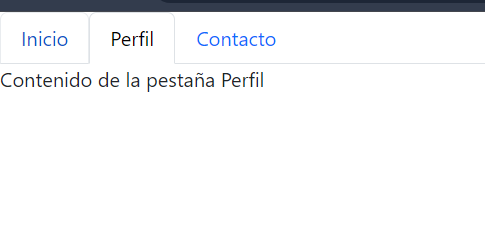
Los Navs y Tabs en Bootstrap se componen de varias partes:

**Nav (navegación):** Contenedor principal que alberga las pestañas.

**Tab (pestaña):** Cada pestaña es un elemento interactivo que muestra un contenido específico.

**Contenido de Tab:** Contenido asociado a cada pestaña.



****

#### Componentes de Navs y Tabs

* **.nav**: Clase base para el contenedor de navegación.
* **.nav-tabs**: Clase que indica que los elementos de navegación son pestañas.
* **.nav-link**: Clase para los enlaces de navegación dentro de los Navs.
* **.tab-content**: Contenedor del contenido de las pestañas.
* **.tab-pane**: Clase para cada sección de contenido asociada a una pestaña.
* **.fade**: Clase opcional para agregar transiciones de desvanecimiento a las pestañas al cambiar entre ellas.
* **.show** y .active: Clases para marcar la pestaña activa y su contenido visible.

#### Uso Avanzado

##### Alineación de Navs

Puedes alinear horizontalmente o verticalmente los Navs utilizando las clases de Bootstrap:

**Horizontal**: Por defecto, los Navs se alinean horizontalmente.

<ul class="nav nav-tabs"> <!-- Navs aquí --> </ul>

Vertical: Usa la clase .flex-column para alinearlos verticalmente.

<ul class="nav nav-tabs flex-column"> <!-- Navs aquí --> </ul>

* **Carousel:**

Un Carousel, también conocido como carrusel o slider, es un componente interactivo que permite mostrar contenido en forma de diapositivas que se desplazan horizontalmente. Es ideal para presentar imágenes, anuncios, productos destacados, testimonios, entre otros, de manera dinámica y visualmente atractiva.

#### Estructura Básica del Carousel

Un Carousel en Bootstrap se compone principalmente de tres partes:

1. **.carousel**: Contenedor principal que envuelve todo el Carousel.
2. **.carousel-inner**: Contenedor que alberga las diapositivas (carousel-item).
3. **.carousel-item**: Cada diapositiva individual del Carousel.

Además, se utilizan controles (carousel-control-prev y carousel-control-next) y indicadores (carousel-indicators) para navegar entre las diapositivas.

#### Componentes del Carousel

* **.carousel**: Clase base del contenedor principal del Carousel.
* **.carousel-inner**: Clase que envuelve las diapositivas (carousel-item).
* **.carousel-item**: Clase para cada diapositiva del Carousel.
* **.carousel-control-prev y .carousel-control-next**: Flechas de control para navegar hacia la diapositiva anterior y siguiente.
* **.carousel-indicators**: Indicadores para mostrar el número y la posición de cada diapositiva.

#### Funcionamiento

* **Auto Play**: El Carousel puede configurarse para avanzar automáticamente (data-bs-ride="carousel").
* **Controles**: Se pueden usar flechas (carousel-control-prev y carousel-control-next) para navegar manualmente entre las diapositivas.
* **Indicadores**: Muestran el número de diapositiva actual y permiten navegar directamente a una diapositiva específica.

#### Opciones Avanzadas

##### Intervalo de Tiempo

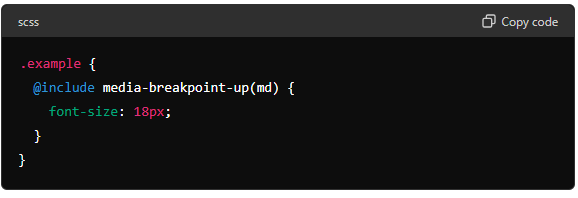
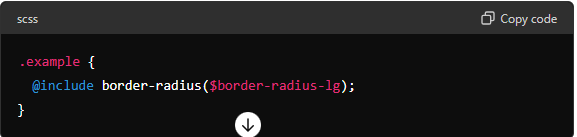
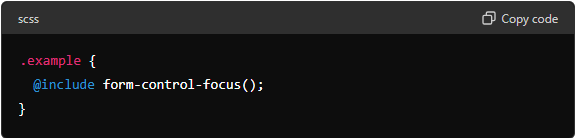
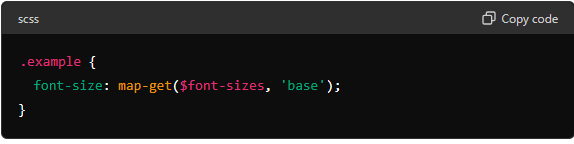
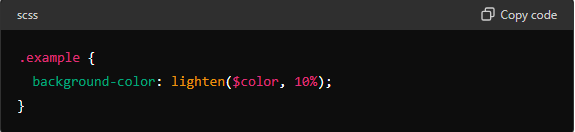
Puedes ajustar el tiempo de transición entre diapositivas usando el atributo data-bs-interval. El valor predeterminado es 5000 (5 segundos).

4 **COMPONENTES DE BOOTSTRAP**

* **Tematización con variables SASS**
  + La "tematización con variables SASS en Bootstrap" se refiere al proceso de personalización de los estilos de Bootstrap utilizando las variables de SASS (Syntactically Awesome Style Sheets). Bootstrap utiliza SASS como preprocesador CSS, lo que permite definir y modificar fácilmente variables que controlan aspectos como colores, tamaños de fuente, espaciados y más en todo el framework
  + Los pasos básicos para la tematización de Bootstrap usando variables son:

Incluir **Bootstrap en tu proyecto**: Asegúrate de tener Bootstrap instalado y configurado en tu proyecto. Puedes hacerlo a través de npm, yarn, o mediante la descarga de archivos desde el sitio web de Bootstrap.

Modificar **variables**: Bootstrap proporciona un archivo \_variables. scss (o similar) donde se definen todas las variables SASS utilizadas en el framework. Este archivo es crucial para personalizar los estilos. Puedes encontrar este archivo en la carpeta scss/ o sass/ dentro de la estructura de carpetas de Bootstrap

* + **Modificar colores**: Una de las personalizaciones más comunes es cambiar los colores de tu tema. En el archivo \_variables. scss, encontrarás variables como $primary, $secondary, $success, etc., que puedes ajustar según tus necesidades de diseño.
  + **Modificar tipografías y tamaños**: Las variables como $font-family-base, $font-size-base, $line-height-base, etc., te permiten ajustar la tipografía y los tamaños de fuente en todo el proyecto.
* **Compilar y usar**: Después de realizar tus modificaciones en las variables SASS, necesitas compilar tu archivo SASS en un archivo CSS que luego puedes incluir en tu proyecto. Puedes compilar usando herramientas como node-sass, sass, o integrando el proceso de compilación en tu flujo de trabajo de desarrollo
* **Personalización adicional**: Además de las variables básicas, Bootstrap también ofrece muchas otras opciones de personalización avanzada que puedes explorar y ajustar según tus necesidades específicas, como los tamaños de los espaciados, los radios de borde, los estilos de los botones, etc.
* Al utilizar variables SASS, puedes mantener la coherencia en tu tema y hacer ajustes globales de manera eficiente. Esto es particularmente útil cuando trabajas en proyectos grandes o cuando necesitas adaptar Bootstrap a los requisitos visuales de una marca específica
* ***Uso de mixins y funciones:***
* **Mixins**
  + Los mixins en SASS son bloques de código reutilizables que contienen un conjunto de declaraciones CSS. En Bootstrap, los mixins son utilizados para generar estilos específicos de componentes o para aplicar estilos con configuraciones predeterminadas.
  + ***@include media-breakpoint-up:*** Este mixin permite aplicar estilos solo a partir de ciertos puntos de quiebre (breakpoints) definidos en las variables de Bootstrap, como sm, md, lg, xl. Por ejemplo:
  + **@include border-radius**: Para aplicar bordes redondeados con radios específicos de acuerdo con las variables de Bootstrap:
  + **@include form-control-focus**: Aplica estilos de foco a los controles de formulario: 
  + **FUNCIONES:**
  + Bootstrap también utiliza funciones SASS para realizar cálculos y devolver valores basados en las variables y configuraciones definidas. Estas funciones son útiles para generar estilos dinámicos y responder a las necesidades del diseño de manera más flexible.
    1. **map-get**: Para obtener un valor específico de un mapa definido en las variables de Bootstrap: 
    2. **theme-color-level**: Para generar un color basado en el nivel de contraste: 
    3. **lighten / darken**: Para ajustar la luminosidad de un color: 
* En tu proyecto, puedes aprovechar estos mixins y funciones para simplificar la creación de estilos personalizados que se integren con la estética general de Bootstrap. Esto no solo mejora la consistencia visual, sino que también facilita el mantenimiento y la adaptación de estilos a medida que evoluciona el diseño y los requisitos del proyecto.
* Para utilizar mixins y funciones en tu proyecto Bootstrap, asegúrate de estar utilizando un entorno de desarrollo que admita la compilación de archivos SASS, como node-sass, sass, o configurando un flujo de trabajo con herramientas como Webpack o Gulp que incluyan soporte para preprocesadores SASS.
* Principio del formulario
* Final del formulario
* ***Creación de temas personalizados:***

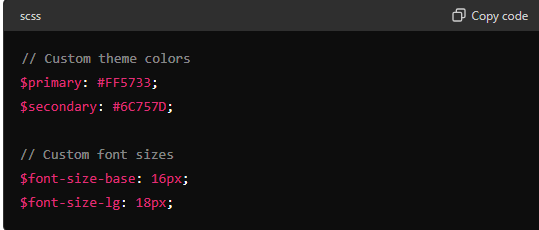
 Configuración **inicial**:

* Asegúrate de tener Bootstrap instalado en tu proyecto. Puedes instalarlo a través de npm, yarn, o descargando los archivos desde el sitio web oficial.

 **Modificación de variables SASS**:

* Bootstrap utiliza variables SASS para definir sus estilos. El archivo principal donde se encuentran estas variables es generalmente \_variables.scss.
* Copia este archivo a tu proyecto y renómbralo según tu preferencia, por ejemplo, custom-variables.scss

.

* Personalización de variables:
  + Abre custom-variables.scss y modifica las variables según el nuevo tema que deseas crear. Algunas variables comunes que puedes ajustar incluyen colores ($primary, $secondary, etc.), tamaños de fuente, espacios ($spacing), radios de borde ($border-radius), y más. 
* **Compilación de SASS**:
  + Utiliza una herramienta de compilación SASS como node-sass, sass, o un flujo de trabajo con Webpack o Gulp para compilar tus archivos SASS personalizados (custom-variables.scss) en un archivo CSS
  + **Inclusión en tu proyecto**:
* Incluye el archivo CSS compilado en tu proyecto, reemplazando el CSS de Bootstrap base. Puedes hacer esto mediante un archivo de entrada principal que importe tu archivo de variables personalizadas y luego compile todo junto.
* **Prueba y ajuste**:
* Aplica el tema personalizado en tu proyecto y realiza pruebas exhaustivas para asegurarte de que todos los estilos se apliquen correctamente y se vean como se espera.

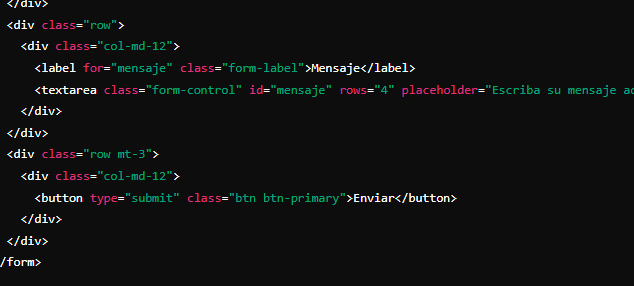
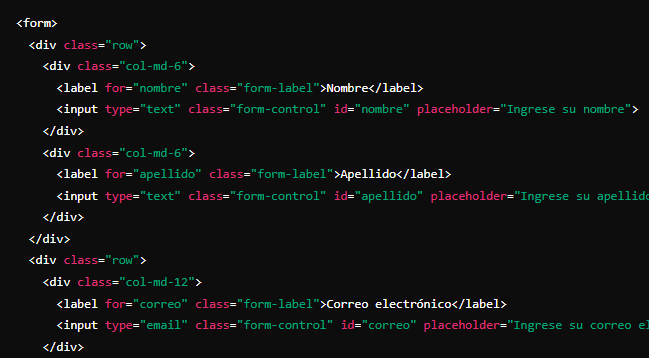
### Consideraciones adicionales:

* **Modificación avanzada**: Además de las variables básicas, Bootstrap ofrece una serie de opciones de personalización más avanzadas, como mixins y funciones, que puedes utilizar para ajustar componentes específicos o crear estilos más complejos.
* **Actualizaciones futuras**: Ten en cuenta que si decides actualizar Bootstrap en el futuro, es posible que necesites revisar y ajustar tus personalizaciones para que sean compatibles con la nueva versión del framework.
* **Documentación y soporte**: Consulta la documentación oficial de Bootstrap para obtener detalles adicionales sobre cómo personalizar el framework y aprovechar al máximo sus características.

Crear un tema personalizado en Bootstrap te permite no solo adaptar el diseño visual a tus necesidades específicas, sino también mantener la coherencia y el estilo en todo tu proyecto web.

* **Formularios Avanzados**

**● Repaso de formularios básicos:**

* En Bootstrap, los formularios se construyen utilizando la estructura de filas (row) y columnas (col). Esto permite que los formularios sean completamente responsivos y se adapten a diferentes tamaños de pantalla. 

### Explicación de los elementos clave:

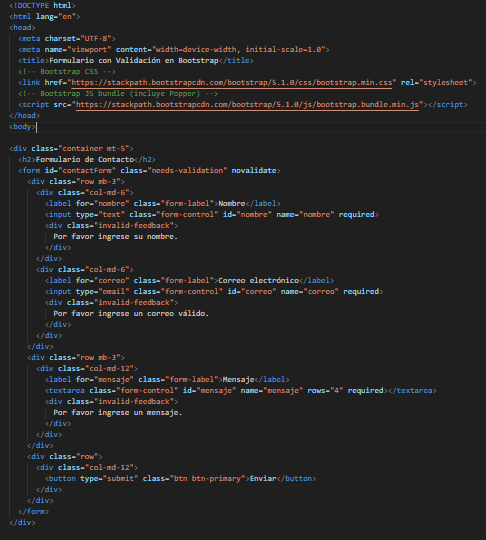
* **.form-control**: Clase de Bootstrap que se aplica a los elementos de entrada (input y textarea) para darles estilos de formulario consistentes.
* **. form-label**: Clase opcional que puedes utilizar para etiquetas de formulario (label). No es parte de Bootstrap, pero puedes usarla para personalizar estilos si es necesario.
* **. row y. col-md-#**: Utilizados para crear una disposición de filas y columnas responsivas. En este ejemplo, cada campo de formulario (input, textarea) está contenido en una columna de tamaño medio (col-md-#), ocupando 6 columnas en pantallas medianas (col-md-6).
* **. btn y .btn-primary**: Clases de Bootstrap para estilizar los botones. .btn-primary aplica el color de botón principal según la configuración de variables de Bootstrap.

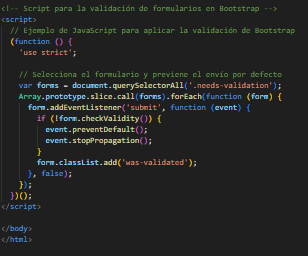
### **Opciones adicionales y características:**

* **Grupos de formulario (form-group)**: Aunque no se ven explícitamente en el ejemplo, en Bootstrap 4 y versiones anteriores, se usaba la clase. form-group para agrupar etiquetas (label) y controles (input, textarea). En Bootstrap 5, se utiliza de manera más automática, pero puedes aún usarla para ajustar el diseño.
* **Elementos de formulario especiales**: Bootstrap también proporciona clases y estilos para manejar selectores (select), grupos de botones (input tipo radio y checkbox), y otros elementos de formulario de manera consistente.
* **Validación de formularios**: Bootstrap ofrece clases específicas y estilos para mostrar mensajes de error y éxito en los formularios validados mediante JavaScript. Puedes encontrar más detalles en la documentación oficial de Bootstrap.

**● Formularios con validación:**

* Para crear un formulario con validación en Bootstrap, puedes utilizar las funcionalidades integradas de Bootstrap 5 para manejar la validación de formularios mediante JavaScript y estilos CSS específicos. A continuación, te guiaré a través de los pasos para implementar un formulario con validación básica:

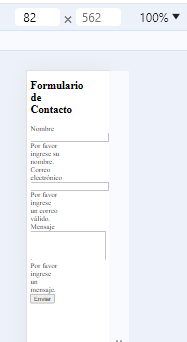
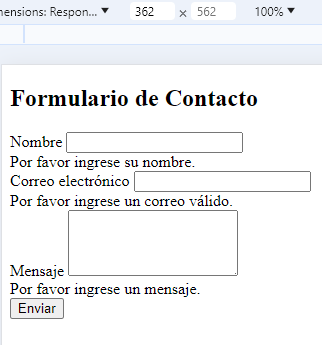




* **HTML del formulario**:
  + El formulario está estructurado usando las clases de Bootstrap para la cuadrícula (row y col-md-#).
  + Cada campo de entrada (input y textarea) tiene la clase form-control para estilizarlo como un campo de formulario de Bootstrap.
  + Se utiliza la clase needs-validation en el formulario para activar la validación de Bootstrap.
* **Atributos de validación HTML5**:
  + Se utiliza el atributo required en los campos de entrada y textarea para requerir que se completen antes de enviar el formulario.
* **Mensajes de retroalimentación de validación**:
  + Se utiliza la clase invalid-feedback junto con la propiedad required en los elementos para mostrar mensajes de error personalizados cuando los campos no son válidos.
* **Script de validación de Bootstrap**:
  + Se incluye un script JavaScript que maneja la validación del formulario usando las funcionalidades de Bootstrap. Este script se encarga de verificar si el formulario es válido antes de enviarlo y muestra los mensajes de error según sea necesario.
* **Bootstrap CSS y JS**:
  + Se incluye Bootstrap CSS y Bootstrap JS bundle (incluyendo Popper.js) desde un CDN para aplicar estilos y funcionalidades de forma rápida y sencilla.

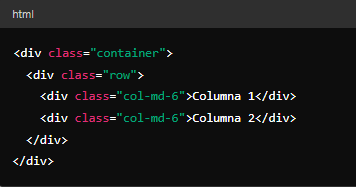
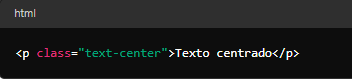
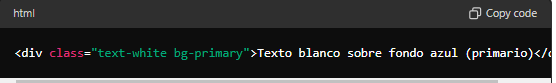
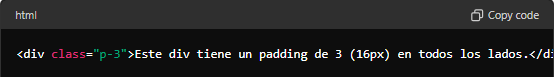
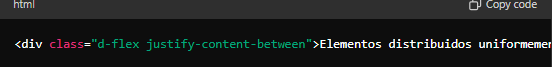
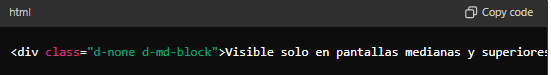
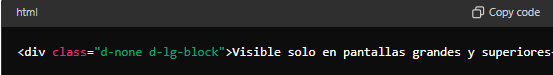
**Funcionamiento del ejemplo:**

* Cuando intentas enviar el formulario, Bootstrap verifica automáticamente que todos los campos requeridos (required) estén completos.
* Si un campo no cumple con los requisitos de validación (como un correo electrónico válido), se mostrará un mensaje de error personalizado.
* El formulario no se enviará hasta que todos los campos requeridos sean válidos.
* Formularios responsivos y accesibles

**6. Utilidades y Helpers**

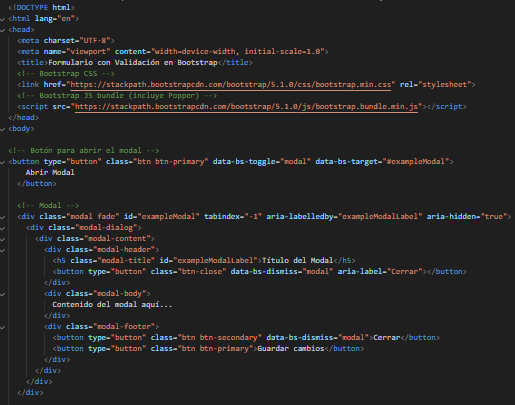
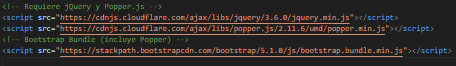
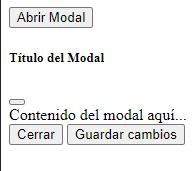
**● Repasodeutilidades básicas:**

* **Sistema de cuadrícula (Grid System)**:
* El sistema de cuadrícula de Bootstrap permite dividir la página en filas (rows) y columnas (cols). Es fundamental para crear diseños responsivos.
* 
* **Tipografía (Typography)**:
* Clases para estilizar texto y mejorar la legibilidad
  + **Color (Colors)**:
* Clases para cambiar el color del texto y del fondo. 
* ***Utilidades avanzadas:***
  + **Espaciado (Spacing)**:
    - Bootstrap proporciona clases para manejar el espaciado (margin y padding) de manera rápida y consistente. 
* **Flexbox (Flexbox Utilities)**:
  + Bootstrap utiliza Flexbox para controlar la distribución y alineación de elementos de manera más flexible. 
* **Visibilidad (Visibility)**:
* Clases para controlar la visibilidad de elementos en diferentes tamaños de pantalla. 
* **Display (Display Utilities)**:
* Clases para cambiar la visibilidad y el comportamiento de display de elementos en diferentes tamaños de pantalla. 

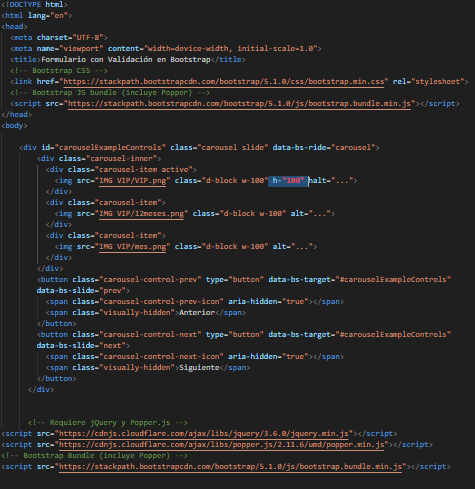
***7. Bootstrap con JavaScript:***

● **Uso de plugins de JavaScript de Bootstrap;**

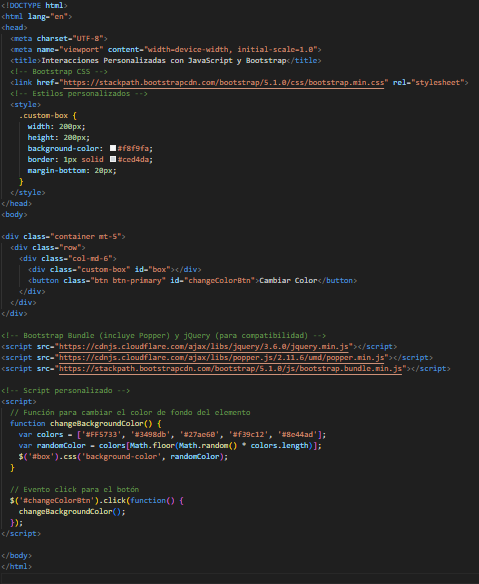
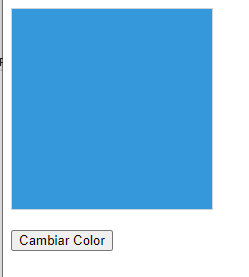
#### **Modal**

* El modal de Bootstrap es una ventana emergente que se utiliza para mostrar contenido como mensajes, formularios o cualquier otro contenido adicional.
* 
* 
* 

#### **Carousel (Carrusel)**

* El carrusel de Bootstrap permite mostrar una serie de imágenes o contenido en un slider interactivo
* 
* 

***● Creación de interacciones personalizadas:***

* + Crear interacciones personalizadas con JavaScript y Bootstrap te permite añadir funcionalidades específicas y personalizadas a tu sitio web, aprovechando las capacidades del framework para el diseño responsivo y la estructura de componentes. Aquí te guiaré a través de un ejemplo básico de cómo puedes crear una interacción personalizada utilizando JavaScript y Bootstrap:
  + En este ejemplo, crearemos una funcionalidad donde un botón cambia dinámicamente el color de fondo de un elemento de manera interactiva
  + 
  + 

1. **HTML y Bootstrap**:
   * Se utiliza Bootstrap para el diseño responsivo y los estilos básicos de los componentes (container, row, col-md-6, btn, etc.).
2. **Estilos personalizados**:
   * Se define un estilo CSS (custom-box) para el elemento div (#box) donde cambiará el color de fondo.
3. **Script JavaScript**:
   * Se define una función changeBackgroundColor() que cambia dinámicamente el color de fondo del elemento #box a un color aleatorio de una lista predefinida.
4. **Evento click**:
   * Se utiliza jQuery para manejar el evento click del botón #changeColorBtn. Cuando se hace clic en el botón, se llama a la función changeBackgroundColor() para cambiar el color de fondo del elemento #box.

### ***Personalización y Ampliación:***

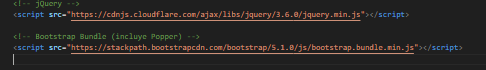
* **Utilización de otros eventos**: Puedes agregar más interacciones utilizando otros eventos como mouseover, keydown, submit, etc., según tus necesidades.
* **Integración con otros componentes de Bootstrap**: Puedes combinar esta lógica con otros componentes de Bootstrap como modales, formularios, o carruseles para crear experiencias más complejas y dinámicas.
* **Animaciones**: Puedes agregar efectos de transición y animaciones CSS para mejorar la experiencia del usuario al cambiar propiedades visuales como colores o tamaños.

***● Integración con otras librerías de JavaScript***

### **Integración con jQuery**

Bootstrap tradicionalmente dependía de jQuery para algunos de sus componentes y funcionalidades interactivas. Aunque Bootstrap 5 está migrando hacia el uso de JavaScript nativo, todavía es compatible con jQuery para ciertos casos.

Ejemplo básico de integración con jQuery**:**





### **Integración con Vue.js**

Vue.js es un framework progresivo de JavaScript para construir interfaces de usuario. Puedes combinar Vue.js con Bootstrap para manejar de manera efectiva la lógica de la interfaz de usuario y los componentes interactivos.

Ejemplo básico de integración con Vue.js:





### **Integración con React**

React es una biblioteca popular para construir interfaces de usuario. Puedes integrar Bootstrap con React para crear aplicaciones web rápidas y escalables.

Ejemplo básico de integración con React:



