

Tema 4: Bootstrap

Semana 14: Botones, tarjetas, carousel, formularios, etc.



BLOQUE 3: Bootstrap

Módulo: Desarrollo de Interfaces Web

Ciclo: Desarrollo de Aplicaciones Web

Curso: 2023 – 2024

Profesora: Rosa Medina

Índice

1. Componentes responsive.....	3
2. Botones.....	3
2.1. Button tags.....	3
2.2. Botones con bordes.....	4
3. Grupo de botones.....	4
3.1. Anidamiento (nesting).....	4
3.2. Botones verticales – por aquí.....	5
4. Tarjetas - Cards.....	5
4.1. Tipos de contenidos.....	5
4.1.1. Bloques.....	5
4.1.2. Títulos, texto y enlaces.....	5
4.1.3. Imágenes.....	6
4.1.4. Grupos de cards.....	6
5. Carousel.....	7
6. Collapse.....	8
7. Desplegables.....	9
8. Formularios.....	10
9. Navs y tabs.....	11
10. Navbar.....	12

1. Componentes responsive

Bootstrap incluye un conjunto de componentes para facilitarnos el diseño de nuestra web y hacerlo mucho más amigable. Estos componentes entre otros son:

- Botones
- Grupo de botones
- Tarjetas
- Carousel
- Collapse
- Desplegables
- Formualrios
- Grupo de inputs
- Navs
- Navbar

2. Botones

Bootstrap tiene varios estilos de [botones](#) predefinidos, cada uno con su propia utilidad (botones.html):

```
<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>
<button type="button" class="btn btn-link">Link</button>
```

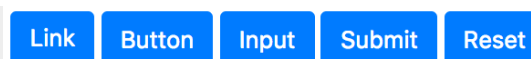


2.1. Button tags

Las clases `.btn` se usan con el elemento `<button>`. Sin embargo, podemos usar estas clases con el elemento HTML `<a>` o `<input>` (aunque algunos navegadores aplican una ligera diferencia de renderizado).

Cuando usamos una clase `.btn` en un elemento `<a>` estamos usando un hipervínculo en la página más que un enlace hacia una nueva página o sección, este enlace debería de tener un `role="button"` para saber el propósito que tiene por ejemplo a la hora de leer información en una pantalla para sordos. Ejemplo botones2.html:

```
<a class="btn btn-primary" href="#" role="button">Link</a>
<button class="btn btn-primary" type="submit">Button</button>
<input class="btn btn-primary" type="button" value="Input">
<input class="btn btn-primary" type="submit" value="Submit">
<input class="btn btn-primary" type="reset" value="Reset">
```



2.2. Botones con bordes

Estos botones tienen la peculiaridad de que el fondo es blanco, se mantiene el color del borde del botón y las letras tienen el color del tipo de botón indicado (clase `.btn-outline-*`). Ejemplo botones3.html:

```
<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>
```



Para ver más funcionalidades con los botones consultad [aquí](#):

3. Grupo de botones

Bootstrap nos ofrece diferentes formas de [agrupar botones](#), veamos un ejemplo básico usando las clases `.btn` y `.btn-group`. Ejemplo gBotones.html

```
<div class="btn-group" role="group" aria-label="Basic example">
  <button type="button" class="btn btn-secondary">Izquierda</button>
  <button type="button" class="btn btn-secondary">Medio</button>
  <button type="button" class="btn btn-secondary">Derecha</button>
</div>
```

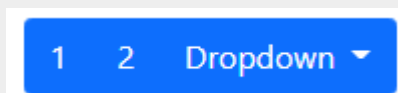


3.1. Anidamiento (nesting)

Lo usaremos cuando queremos tener un grupo de botones mezclado con un desplegable. Ejemplo gBotones5.html:

```
<div class="btn-group" role="group" aria-label="Button group with nested dropdown">
  <button type="button" class="btn btn-primary">1</button>
  <button type="button" class="btn btn-primary">2</button>

  <div class="btn-group" role="group">
    <button id="btnGroupDrop1" type="button" class="btn btn-primary dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">
      Dropdown
    </button>
    <ul class="dropdown-menu" aria-labelledby="btnGroupDrop1">
      <li><a class="dropdown-item" href="#">Dropdown link</a></li>
      <li><a class="dropdown-item" href="#">Dropdown link</a></li>
    </ul>
  </div>
</div>
```



3.2. Botones verticales – por aquí

Si queremos que alguno de los ejemplos anteriores en lugar de tener los botones en horizontal (por defecto) estén en vertical, tendríamos que agruparlos usando la clase: `.btn-group-vertical`. Ejemplo:

```
<div class="btn-group-vertical" role="group" aria-label="Vertical button group">
  ...
</div>
```

Más información [aquí](#).

4. Tarjetas - Cards

Una [card](#) es un contenedor flexible y estensible. Incluye opciones para las encabezados y los pies, una amplia variedad de contenido, colores de fondo y numerosas opciones más de visualización.

Las cards están hechas con un pequeño maquetado pero todo se puede personalizar. Están contruidos con flexbox, de forma que ofrecen una fácil alineación y se mezclan bien con otros componentes de Bootstrap. Veamos un ejemplo básico de un card, card.html.

```
<div class="card" style="width: 20rem;">
  
  <div class="card-body">
    <h4 class="card-title">Nadal, Nº 1</h4>
    <p class="card-text">Ejemplo rápido de cómo crear nuestra propia card, añadiendo
    una imagen, un título y algo de contenido</p>
    <a href="#" class="btn btn-primary">Seguir leyendo</a>
  </div>
</div>
```



4.1. Tipos de contenidos

Las cards tienen una amplia variedad de contenido, incluyendo imágenes, texto, listados, enlaces y mucho más, veamos algún ejemplo:

4.1.1. Bloques

Ejemplo con bordes en la card, card2.html

```
<div class="card">
  <div class="card-body">
    Ejemplo de texto con bordes en la card.
  </div>
</div>
```

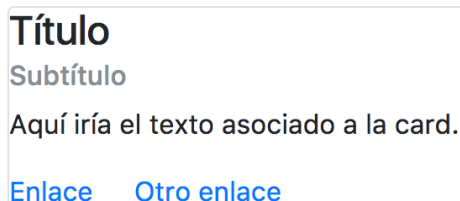
Ejemplo de texto con bordes en la card.

4.1.2. Títulos, texto y enlaces

Para añadirle un título a la card añadiremos al `<h*>` la clase `.card-title`, del mismo modo para los enlaces `<a>`, usaremos: `.card-link`.

Para los subtítulos usaremos en los `<h>` la clase `.card-subtitle`, y si el subtítulo se encuentra dentro de la card y queremos alinearlos con el título usaremos: `.card-block`. Ejemplo: cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como extensiones de JavaScript adicionales. A diferencia de muchos frameworks web, solo se ocupa del desarrollo front-end. Ejemplo `card3.html`

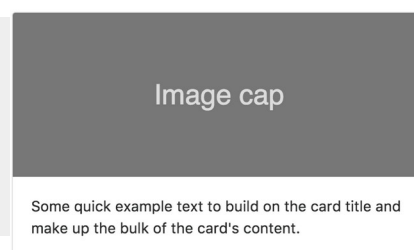
```
<div class="card" style="width: 20rem;">
  <div class="card-body">
    <h4 class="card-title">Título</h4>
    <h6 class="card-subtitle mb-2 text-muted">Subtítulo</h6>
    <p class="card-text">Aquí iría el texto asociado a la card</p>
    <a href="#" class="card-link">Enlace</a>
    <a href="#" class="card-link">Otro enlace</a>
  </div>
</div>
```



4.1.3.Imágenes

Con la clase `.card-img-top` nos sitúa la imagen en la parte de arriba de la card. Con la clase `.card-text` podemos añadir texto a la imagen. Ejemplo [card4.html](#):

```
<div class="card" style="width: 20rem;">
  
  <div class="card-body">
    <p class="card-text">Some quick example text to build on
the card title and make up the bulk of the card's content.</p>
  </div>
</div>
```

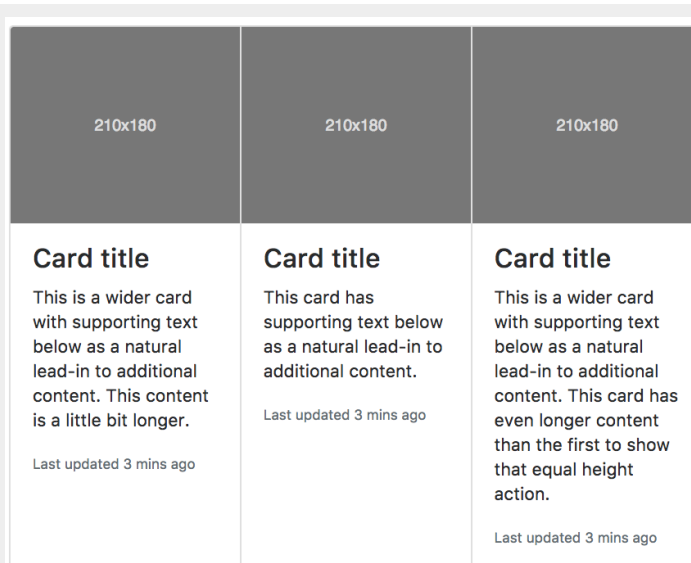


Para consultar más ejemplos sobre las cards con imágenes, [aquí](#).

4.1.4. Grupos de cards

Usaremos los grupos de Cards como un único elemento de modo que cada Card tenga mismo tamaño tanto en anchura como en altura usaremos `display:flex`; para conseguir este tamaño uniforme. Ejemplo `card10.html`:

<pre><div class="card-group"> <div class="card"> <div class="card-block"> <h4 class="card-title">Card title</h4> <p class="card-text">This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.</p> <p class="card-text"><small class="text- muted">Last updated 3 mins ago</small></p> </div> </div> <div class="card"> <div class="card-block"> <h4 class="card-title">Card title</h4> <p class="card-text">This card has supporting text below as a natural lead-in to additional content.</p> <p class="card-text"><small class="text-muted">Last updated 3 mins ago</small></p> </div></pre>			
	<table><tr><td data-bbox="603 1572 738 1800"><p>Card title</p><p>This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.</p><p>Last updated 3 mins ago</p></td><td data-bbox="738 1572 949 1800"><p>Card</p><p>This ca support as a na additio Last upda</p></td></tr></table>	<p>Card title</p> <p>This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.</p> <p>Last updated 3 mins ago</p>	<p>Card</p> <p>This ca support as a na additio Last upda</p>
<p>Card title</p> <p>This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.</p> <p>Last updated 3 mins ago</p>	<p>Card</p> <p>This ca support as a na additio Last upda</p>		



```

</div>
<div class="card">
  
  <div class="card-block">
    <h4 class="card-title">Card title</h4>
    <p class="card-text">This is a wider card with supporting text below as a natural lead-in to additional content. This card has even longer content than the first to show that equal height action.</p>
    <p class="card-text"><small class="text-muted">Last updated 3 mins ago</small></p>
  </div>
</div>
</div>

```

Si quisiéramos añadirle un footer a cada una de las cards, lo que haremos será añadirle a cada div su footer de la siguiente forma:

```

<div class="card">
  ...
  <div class="card-footer">
    <small class="text-muted">Last updated 3 mins ago</small>
  </div>
</div>

```

5. Carousel

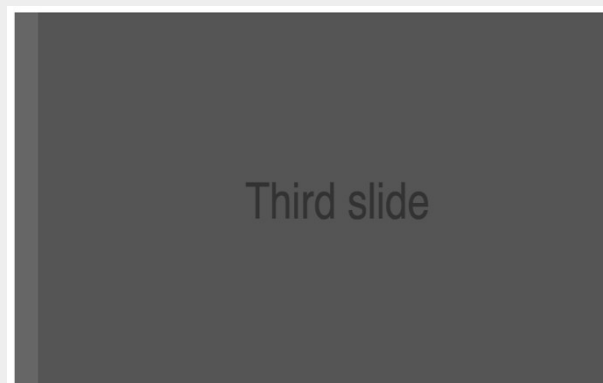
Con respecto al [carousel](#), debemos tener en cuenta que tenemos que darle una única id al elemento que contenga la clase `.carousel` con el objetivo de gestionar y controlar los controles opcionales que tenemos, especialmente si estamos usando varios carousels.

Ejemplo de sólo deslizamiento, se emplean las clases `.d-block` y `.w-100`, veamos un ejemplo `carousel.html`:

```

<div id="carouselExampleSlidesOnly" class="carousel slide" data-bs-ride="carousel">
  <div class="carousel-inner">
    <div class="carousel-item active">
      
    </div>
    <div class="carousel-item">
      
    </div>
    <div class="carousel-item">
      
    </div>
  </div>
</div>

```



Carousel con indicadores de en qué posición se encuentra:

```

<div id="carouselExampleIndicators" class="carousel slide" data-bs-ride="carousel">
  <div class="carousel-indicators">
    <button type="button" data-bs-target="#carouselExampleIndicators" data-bs-slide-to="0"

```

```

class="active" aria-current="true" aria-label="Slide 1"></button>
  <button type="button" data-bs-target="#carouselExampleIndicators" data-bs-slide-to="1" aria-
label="Slide 2"></button>
  <button type="button" data-bs-target="#carouselExampleIndicators" data-bs-slide-to="2" aria-
label="Slide 3"></button>
</div>
<div class="carousel-inner">
  <div class="carousel-item active">
    
  </div>
  <div class="carousel-item">
    
  </div>
  <div class="carousel-item">
    
  </div>
</div>
<button class="carousel-control-prev" type="button" data-bs-target="#carouselExampleIndicators"
data-bs-slide="prev">
  <span class="carousel-control-prev-icon" aria-hidden="true"></span>
  <span class="visually-hidden">Previous</span>
</button>
<button class="carousel-control-next" type="button" data-bs-target="#carouselExampleIndicators"
data-bs-slide="next">
  <span class="carousel-control-next-icon" aria-hidden="true"></span>
  <span class="visually-hidden">Next</span>
</button>
</div>

```



Más ejemplos [aquí](#)

6. Collapse

Esta [propiedad](#) nos permite mostrar/ocultar otro elemento html, para ello las clases que disponemos son:

- .collapse: oculta el contenido
- .collapsing: se aplica durante las transiciones
- .collapse.show muestra el contenido

Podemos usar un enlace con el atributo href o un botón con el data-target, en ambos casos el atributo data-toggle="collapse" es obligatorio

```

<p>
  <a class="btn btn-primary" data-toggle="collapse" href="#collapseExample" role="button" aria-expanded="false" aria-
controls="collapseExample">
    Link with href
  </a>
  <button class="btn btn-primary" type="button" data-toggle="collapse" data-target="#collapseExample" aria-expanded="false"
aria-controls="collapseExample">

```



```

    Button with data-target
  </button>
</p>
<div class="collapse" id="collapseExample">
  <div class="card card-body">
    Anim pariatur cliche reprehenderit, enim eiusmod high life accusamus terry richardson ad squid. Nihil anim keffiyeh helvetica, craft beer labore wes anderson cred nesciunt sapiente ea proident.
  </div>
</div>

```

Antes	Después
<div> Link with href <a data-target="#" href="#">Button with data-target </div>	<div> Link with href <a data-target="#" href="#">Button with data-target </div> <div> Anim pariatur cliche reprehenderit, enim eiusmod high life accusamus terry richardson ad squid. Nihil anim keffiyeh helvetica, craft beer labore wes anderson cred nesciunt sapiente ea proident. </div>

Más ejemplos de collapse y utilidades [aquí](#).

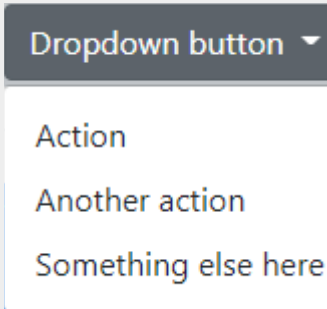
7. Desplegables

Ejemplo de desplegable con botón, desplegable.html:

```

<div class="dropdown">
  <button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton1"
  data-bs-toggle="dropdown" aria-expanded="false">
    Dropdown button
  </button>
  <ul class="dropdown-menu" aria-labelledby="dropdownMenuButton1">
    <li><a class="dropdown-item" href="#">Action</a></li>
    <li><a class="dropdown-item" href="#">Another action</a></li>
    <li><a class="dropdown-item" href="#">Something else here</a></li>
  </ul>
</div>

```

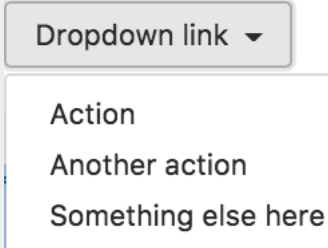


Ejemplo de desplegable con enlace, ejemplo desplegable2.html:

```

<div class="dropdown">
  <a class="btn btn-secondary dropdown-toggle" href="#"
  role="button" id="dropdownMenuLink" data-bs-toggle="dropdown"
  aria-expanded="false">
    Dropdown link
  </a>
  <ul class="dropdown-menu" aria-labelledby="dropdownMenuLink">
    <li><a class="dropdown-item" href="#">Action</a></li>
    <li><a class="dropdown-item" href="#">Another action</a></li>
    <li><a class="dropdown-item" href="#">Something else here</a></li>
  </ul>
</div>

```



Lo mejor de todo en Bootstrap, es que los desplegables los podemos hacer con cualquier variante de botones como vimos en el punto de los botones.

8. Formularios

A continuación, vamos a ver algunos de los tipos de elementos que podemos utilizar en los [formularios](#), es importante tener en cuenta que todos los inputs deben tener el atributo type.

Controles de formularios y tipos de inputs, ejemplo forms2.html

```
<form>
  <div class="mb-3">
    <label for="exampleInputEmail1" class="form-label">Email address</label>
    <input type="email" class="form-control" id="exampleInputEmail1" aria-
describedby="emailHelp">
    <div id="emailHelp" class="form-text">We'll never share your email with anyone else.</div>
  </div>
  <div class="mb-3">
    <label for="exampleInputPassword1" class="form-label">Password</label>
    <input type="password" class="form-control" id="exampleInputPassword1">
  </div>
  <div class="mb-3">
    <label for="exampleColorInput" class="form-label">Color picker</label>
    <input type="color" class="form-control form-control-color" id="exampleColorInput"
value="#563d7c" title="Choose your color">
  </div>
  <div class="mb-3">
    <div class="form-check">
      <input class="form-check-input" type="radio" name="exampleRadios" id="exampleRadios1"
value="option1" checked>
      <label class="form-check-label" for="exampleRadios1">Default radio</label>
    </div>
    <div class="form-check">
      <input class="form-check-input" type="radio" name="exampleRadios" id="exampleRadios2"
value="option2">
      <label class="form-check-label" for="exampleRadios2">Second default radio</label>
    </div>
    <div class="form-check">
      <input class="form-check-input" type="radio" name="exampleRadios" id="exampleRadios3"
value="option3" disabled>
      <label class="form-check-label" for="exampleRadios3">Disabled radio</label>
    </div>
  </div>
  <div class="mb-3">
    <select class="form-select" multiple aria-label="multiple select example">
      <option selected>Open this select menu</option>
      <option value="1">One</option>
      <option value="2">Two</option>
      <option value="3">Three</option>
    </select>
  </div>
  <div class="mb-3">
    <label for="customRange3" class="form-label">Example range</label>
    <input type="range" class="form-range" min="0" max="5" step="0.5" id="customRange3">
  </div>
  <div class="mb-3">
    <label for="exampleDataList" class="form-label">Datalist example</label>
    <input class="form-control" list="datalistOptions" id="exampleDataList" placeholder="Type to
search...">
```

```

<datalist id="datalistOptions">
  <option value="San Francisco">
  <option value="New York">
  <option value="Seattle">
  <option value="Los Angeles">
  <option value="Chicago">
</datalist>
</div>
<button type="submit" class="btn btn-primary">Submit</button>
</form>

```

Email address

We'll never share your email with anyone else.

Password

Color picker

☒ Default radio
☐ Second default radio
☐ Disabled radio

Open this select menu

- One
- Two
- Three

Example range

Datalist example

Submit

9. Navs y tabs

Los [navs](#) se utilizan para navegar a lo largo de nuestra web. Para conseguirlo con Bootstrap debemos utilizar al menos la clase `.nav` que nos va a permitir activar y desactivar los estados.

El componente básico utiliza la clase `.nav` como se ha mencionado, este no incluye el estado `.active` si nosotros no lo ponemos por defecto. En el siguiente ejemplo, podemos ver cómo usar el modelo básico, `navs.html`

```

<ul class="nav">
  <li class="nav-item">
    <a class="nav-link active" href="#">Active</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">Link</a>
  </li>
  <li class="nav-item">

```

Active Link Link Disabled

```

<a class="nav-link" href="#">Link</a>
</li>
<li class="nav-item">
  <a class="nav-link disabled" href="#">Disabled</a>
</li>
</ul>

```

10. Navbar

Antes de nada, vamos a ver cómo funciona el [navbar](#), y lo que necesitamos conocer antes de empezar:

- Navbars require un empaquetado `.navbar` con `.navbar-expand{-sm|-md|-lg|-xl-xxl}` para un responsive collapse y esquema de [clases de color](#).
- Navbars y sus contenidos son fluid por defecto. Por tanto podemos utilizar [contenedores de forma opcional](#) para limitar el ancho horizontal que tienen.
- Debemos usar los [espaciados](#) y utilidades de [flex](#) para controlarlos y alinear los elementos del navbar.
- Navbars son responsive por defecto, pero podemos de forma fácil modificarlos y cambiarlos. El comportamiento responsive depende del plugin Collapse de JavaScript.
- Debemos asegurarnos una accesibilidad usando el element `<nav>`, o si usamos un elemento más genérico como `<div>`, debemos añadir `role="navigation"` en cada navbar para identificarlos de forma explícita, es como una region de referencia para los usuarios que necesitan de una tecnología de asistencia.
- Para indicar el ítem actual en el navbar usaremos `aria-current="page"` para la página activa o `aria-current="true"` para el ítem.

El contenido que soporta es:

- `.navbar-brand` para nuestra compañía, product o nombre de nuestro proyecto.
- `.navbar-nav` para darle toda la altura y anchura de la navegación (incluye el soporte para dropdowns).
- `.navbar-toggler` Para usarlo con el plugin de collapse de Bootstrap y otros comportamientos de navegación.
- Utilidades de Flex y espaciados para el control de formularios y acciones
- `.navbar-text` Para añadir un centrado vertical a las cadenas de texto
- `.collapse.navbar-collapse` Para agrupar y ocultar el contenido del navbar utilizando un breakpoint.
- Y de forma opcional `.navbar-scroll` para establecer un `max-height` y [expandir el contenido del navbar](#).

Vamos a ver un ejemplo de todos los componentes que podemos incluir en un tema responsive del navbar que de forma automática se collapse si no estamos con una resolución lg (grande) y no cabe.

```

<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-
label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarSupportedContent">
      <ul class="navbar-nav me-auto mb-2 mb-lg-0">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Link</a>
        </li>
        <li class="nav-item dropdown">
          <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button" data-
bs-toggle="dropdown" aria-expanded="false">Dropdown</a>
          <ul class="dropdown-menu" aria-labelledby="navbarDropdown">
            <li><a class="dropdown-item" href="#">Action</a></li>
            <li><a class="dropdown-item" href="#">Another action</a></li>
            <li><hr class="dropdown-divider"></li>
            <li><a class="dropdown-item" href="#">Something else here</a></li>
          </ul>
        </li>
        <li class="nav-item">
          <a class="nav-link disabled">Disabled</a>
        </li>
      </ul>
      <form class="d-flex">
        <input class="form-control me-2" type="search" placeholder="Search" aria-
label="Search">
        <button class="btn btn-outline-success" type="submit">Search</button>
      </form>
    </div>
  </div>
</nav>

```

Navbar Home Link Dropdown ▾ Disabled