

BOOTSTRAP

WHAT IS BOOTSTRAP?

- Bootstrap is a free front-end framework for faster and easier web development.
- Bootstrap includes HTML and CSS based design templates for forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plug-in.
- Bootstrap also gives you the ability to easily create responsive designs.

ADVANTAGES OF BOOTSTRAP:

- Easy to use.
- Responsive features.
- Mobile-first approach.
- Browser compatibility.

HISTORY OF BOOTSTRAP:

- Bootstrap was developed by **MARK OTTO & JACOB THORNTON** at twitter and released as an open source product in August 2011 on GitHub.
- Bootstrap originally named **Twitter Blueprint**.
- In June 2014 Bootstrap was the No.1 project on Github!
- Previous Bootstrap Version:

Version	Release date
Bootstrap 2	January 31 2012
Bootstrap 3	August 19 2013
Bootstrap 4	October 29 2014
Bootstrap 5	May 5 2021

BOOTSTRAP 5 CDN:

- If you don't want to download and host Bootstrap 5 yourself, you can include it from a CDN (Content Delivery Network).
- jsDelivr provides CDN support for Bootstrap's CSS and JavaScript.

- CSS(cdn):

<link

**href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css
" rel="stylesheet">**

- Script(cdn):

<script

**src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.m
in.js"></script>**

BOOTSTRAP 5:

- Bootstrap 5 is designed to be responsive to mobile devices. To ensure proper rendering and touch zooming, add the following **<meta>** tag inside the **<head>** element:
<meta name="viewport" content="width=device-width, initial-scale=1">
- The **width=device-width** part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).
- The **initial-scale=1** part sets the initial zoom level when the page is first loaded by the browser.

WHAT IS RESPONSIVE WEB DESIGN?

- Responsive web design is about creating web sites which automatically adjust themselves to look good on all devices, from small phones to large desktops.



RESPONSIVE CONTAINERS

- Extra small (xs) →Portrait phones
- Small (sm) →Landscape phones
- Medium (md) →Tablets
- Large (lg) →Laptops
- Extra large (xl) →Laptops and Desktops
- Extra extra large (xxl)→Laptops and Desktops

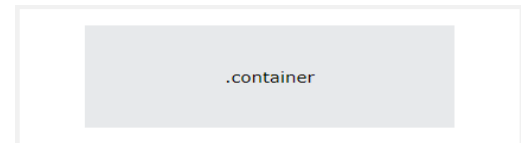
CONTAINERS:

- .container -fixed width container
- .container-fluid -full width container

Example-1:

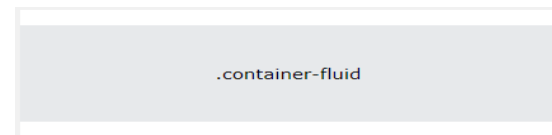
- Container

```
<div class="container">  
  <h1>My First Bootstrap Page</h1>  
  <p>This part is inside a .container class.</p>  
  <p>The .container class provides a responsive fixed width container.</p>  
</div>
```

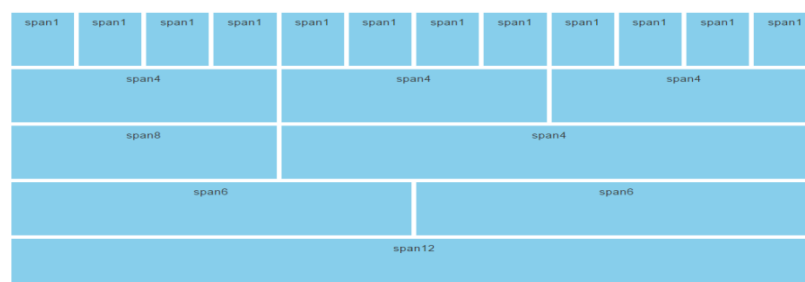


- Container-fluid

```
<div class="container-fluid">  
  <h1>My First Bootstrap Page</h1>  
  <p>This part is inside a .container class.</p>  
  <p>The .container class provides a responsive fixed width container.</p>  
</div>
```



BOOTSTRAP GRIDS:



Example:

```

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

```

FIXED CONTAINER:

- Use the .container class to create a responsive, fixed-width container.
- Its width (max-width) will change on different screen sizes.

	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra Large ≥1200px	XXL ≥1400px
max-width	100%	540px	720px	960px	1140px	1320px

RESPONSIVE CONTAINERS:

- The max-width of the container will change on different screen sizes/viewports.

Class	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra large ≥1200px	XXL ≥1400px
.container-sm	100%	540px	720px	960px	1140px	1320px
.container-md	100%	100%	720px	960px	1140px	1320px
.container-lg	100%	100%	100%	960px	1140px	1320px
.container-xl	100%	100%	100%	100%	1140px	1320px
.container-xxl	100%	100%	100%	100%	100%	1320px

OFFSET:

- Move columns to the right using **.offset-md-*** classes.
- For example, **.offset-md-4** moves **.col-md-4**



➤ Example:

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

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

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

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

</div>

</div>

MARGIN & PADDING:

- m- margin.
- p- padding.
- t- margin-top or padding-top.
- b- margin-bottom or padding-bottom.
- s- margin-left or padding-left.
- e- margin-right or padding-right.
- x- both left & right.
- y- top & bottom.
- 0- eliminate the margin or padding by set.
- The values for margin & padding starts from 1 to 5.
- auto – for classes that set the margin & padding auto.

Example:

```
<div class="container">  
  <div class="row my-5">  
    <div class="col-md-4 mt-4">.col-md-4</div>  
    <div class="col-md-4 ps-auto">.col-md-4 .ms-auto</div>  
  </div>  
</div>
```

TEXT/TYPOGRAPHY:

- Bootstrap 5 uses a default font-size of 1rem (16px by default), and its line-height is 1.5.
- In addition, all <p> elements have margin-top: 0 and margin-bottom: 1rem (16px by default).

➤ **<h1>-<h6>**

- ✓ Bootstrap 5 styles HTML headings (<h1> to <h6>) with a bolder font-weight and a responsive font-size.

- ✓ **Example:**

```
<div class="container">
  <p class="h1">h1 Bootstrap heading</p>
  <p class="h2">h2 Bootstrap heading</p>
  <p class="h3">h3 Bootstrap heading</p>
  <p class="h4">h4 Bootstrap heading</p>
  <p class="h5">h5 Bootstrap heading</p>
  <p class="h6">h6 Bootstrap heading</p>
</div>
```

h1 Bootstrap heading

h2 Bootstrap heading

h3 Bootstrap heading

h4 Bootstrap heading

h5 Bootstrap heading

h6 Bootstrap heading

➤ **Display Headings:**

- ✓ Display headings are used to stand out more than normal headings (larger font-size and lighter font-weight).
- ✓ There are six classes to choose from display-1 to display-6
- ✓ Example:

```
<div class="container mt-3">
  <h1 class="display-1">Display 1</h1>
  <h1 class="display-2">Display 2</h1>
  <h1 class="display-3">Display 3</h1>
  <h1 class="display-4">Display 4</h1>
  <h1 class="display-5">Display 5</h1>
  <h1 class="display-6">Display 6</h1>
</div>
```

Display 1

Display 2

Display 3

Display 4

Display 5

Display 6

➤ **Font Size:**

```
<p class="fs-1">.fs-1 text</p>
<p class="fs-2">.fs-2 text</p>
<p class="fs-3">.fs-3 text</p>
<p class="fs-4">.fs-4 text</p>
<p class="fs-5">.fs-5 text</p>
<p class="fs-6">.fs-6 text</p>
```

.fs-1 text

.fs-2 text

.fs-3 text

.fs-4 text

.fs-5 text

.fs-6 text

➤ **Font Weight:**

- ✓ Quickly change the font-weight or font-style of text with these utilities. font-style utilities are abbreviated as .fst-* and font-weight utilities are abbreviated as .fw-.*.

✓ Example:

```
<p class="fw-bold">Bold text.</p>
<p class="fw-bolder">Bolder weight text
(relative to the parent element).</p>
<p class="fw-semibold">Semibold weight text.</p>
<p class="fw-medium">Medium weight text.</p>
<p class="fw-normal">Normal weight text.</p>
<p class="fw-light">Light weight text.</p>
<p class="fw-lighter">Lighter weight text
(relative to the parent element).</p>
<p class="fst-italic">Italic text.</p>
<p class="fst-normal">Text with normal font style</p>
```

Bold text.

Bolder weight text (relative to the parent element).

Semibold weight text.

Medium weight text.

Normal weight text.

Light weight text.

Lighter weight text (relative to the parent element).

Italic text.

Text with normal font style

➤ **Small:**

- ✓ It is used to create a smaller, secondary text in any heading.
- ✓ Example:

```
<h1>h1 heading <small>secondary text</small></h1>
```

➤ **Mark:**

- ✓ with a yellow background color and some padding.
- ✓ Example:

```
<p>Use the mark element to <mark>highlight</mark> text.</p>
```

➤ **Abbreviation:**

- ✓ With a dotted border bottom and a cursor with question mark on hover.
- ✓ Example:

```
<p>The <abbr title="World Health Organization">WHO</abbr> was
founded in 1948.</p>
```

➤ **Block Quote:**

- ✓ Add the .blockquote class to a <blockquote> when quoting blocks of content from another source. And when naming a source, like "from WWF's website", use the .blockquote-footer class.
- ✓ Example:

```
<blockquote class="blockquote">
```


<p>For 50 years, WWF has been protecting the future of nature.</p>

<footer class="blockquote-footer">From WWF's website</footer>

</blockquote>

➤ **Code:**

✓ Inline snippets of code should be embedded in the code element.

✓ Example:

<p>The following HTML elements: <code>span</code>, <code>section</code>, and <code>div</code> defines a section in a document.</p>

➤ **Keyboard Inputs:**

✓ To indicate input that is typically entered via the keyboard, use the kbd element.

✓ Example:

<p>Use <kbd>ctrl + p</kbd> to open the Print dialog box.</p>

➤ **Text alignment:**

✓ Left aligned:

<p class="text-start">Left-aligned text.</p>

✓ Right aligned:

<p class="text-end">Right-aligned text.</p>

✓ Center aligned:

<p class="text-center">Center-aligned text.</p>

➤ **Text-Transform:**

✓ Uppercase:

<p class="text-uppercase">Lowercased text.</p>

✓ Lowercase:

<p class="text-lowercase">Uppercased text.</p>

✓ Capitalize:

<p class="text-capitalize">Capitalized text.</p>

➤ **Lead:**

✓ Makes a paragraph stand out.

✓ Example:

<p class="lead">This paragraph stands out.</p>

➤ **Text-break:**

✓ Prevents long text from breaking layout.

- ✓ Example:

```
<p class="text-break">
AaBbCcDdEeFfGgHhIiJjKkLlMmNnOoPpQqRrSsTtUuVvWwXxYyZz
</p>
```

➤ **List-inline:**

- ✓ Places all list items on a single line.

- ✓ Example:

```
<ul class="list-inline">
  <li class="list-inline-item">Coffee</li>
  <li class="list-inline-item">Tea</li>
  <li class="list-inline-item">Milk</li>
</ul>
```

➤ **List-unstyled:**

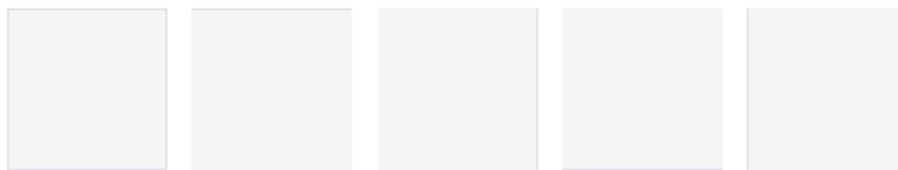
- ✓ Removes the default list-style and left margin on list items.

- ✓ Example:

```
<ul class="list-unstyled">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

BORDER:

- Use border utilities to quickly style the border and border-radius of an element. Great for images, buttons, or any other element.



Example:

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

Subtractive:

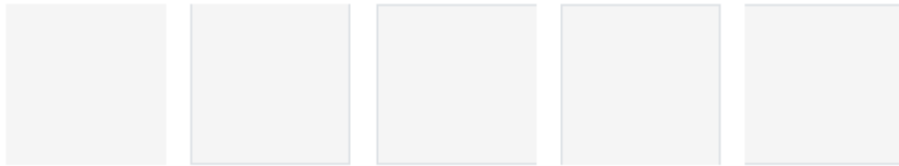
``

``

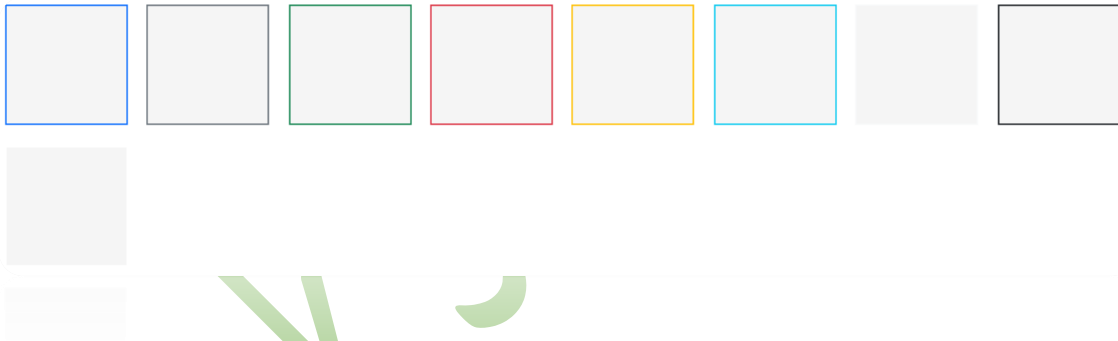
``

``

``



Border color:



``

``

``

``

``

``

``

``

``

BORDER-WIDTH:

```
<span class="border border-1"></span>
```

```
<span class="border border-2"></span>
```

```
<span class="border border-3"></span>
```

```
<span class="border border-4"></span>
```

```
<span class="border border-5"></span>
```



BORDER-RADIUS

```

```

```

```

```

```

```

```

```

```

```

```

```

```



Sizes:

```

```

```

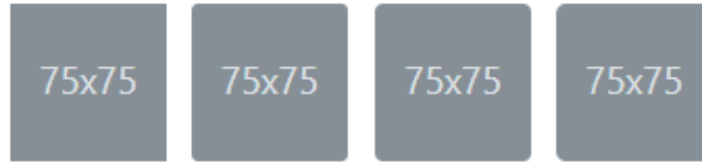
```

```

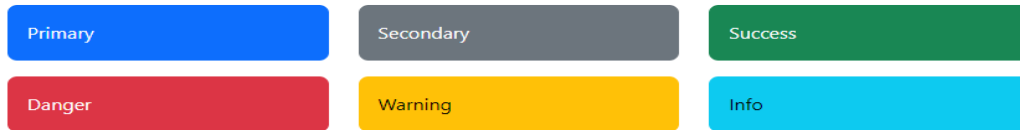
```

```

```



TEXT COLORS:



<p class="text-primary">.text-primary</p>

<p class="text-primary-emphasis">.text-primary-emphasis</p>

<p class="text-secondary">.text-secondary</p>

<p class="text-secondary-emphasis">.text-secondary-emphasis</p>

<p class="text-success">.text-success</p>

<p class="text-success-emphasis">.text-success-emphasis</p>

<p class="text-danger">.text-danger</p>

<p class="text-danger-emphasis">.text-danger-emphasis</p>

<p class="text-warning bg-dark">.text-warning</p>

<p class="text-warning-emphasis">.text-warning-emphasis</p>

<p class="text-info bg-dark">.text-info</p>

<p class="text-info-emphasis">.text-info-emphasis</p>

.text-light

.text-light-emphasis

.text-dark

.text-dark-emphasis

.text-body

.text-body-emphasis

.text-body-secondary

.text-body-tertiary

.text-black

.text-white

.text-black-50

.text-white-50

BACKGROUND COLOR

<div class="p-3 mb-2 bg-primary text-white">.bg-primary</div>

<div class="p-3 mb-2 bg-secondary text-white">.bg-secondary</div>

<div class="p-3 mb-2 bg-success text-white">.bg-success</div>

<div class="p-3 mb-2 bg-danger text-white">.bg-danger</div>

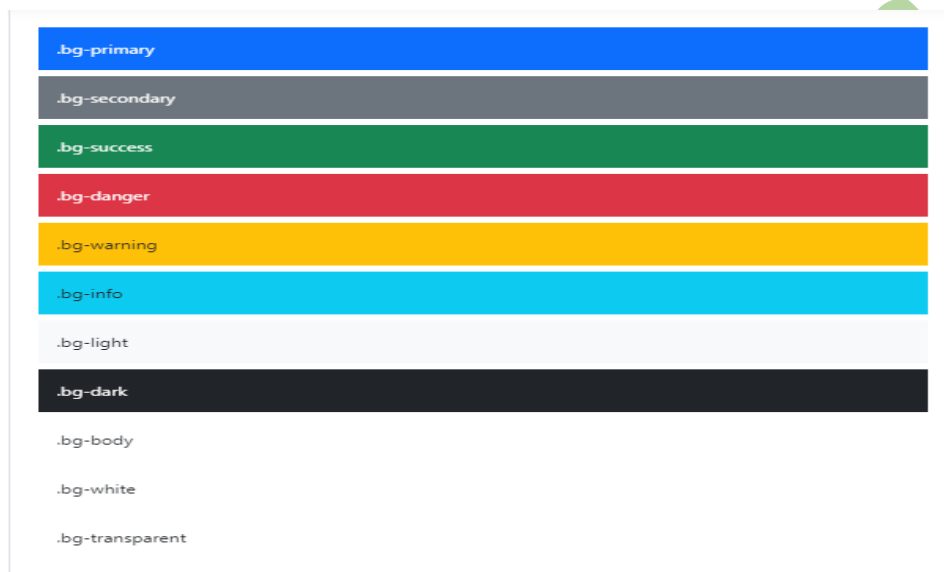
<div class="p-3 mb-2 bg-warning text-dark">.bg-warning</div>

<div class="p-3 mb-2 bg-info text-dark">.bg-info</div>

```

<div class="p-3 mb-2 bg-light text-dark">.bg-light</div>
<div class="p-3 mb-2 bg-dark text-white">.bg-dark</div>
<div class="p-3 mb-2 bg-body text-dark">.bg-body</div>
<div class="p-3 mb-2 bg-white text-dark">.bg-white</div>
<div class="p-3 mb-2 bg-transparent text-dark">.bg-transparent</div>

```



OPACITY:

```

<div class="bg-success p-2 text-white">This is default success </div>
<div class="bg-success p-2 text-white bg-opacity-75">75% opacity </div>
<div class="bg-success p-2 text-dark bg-opacity-50">50% opacity </div>
<div class="bg-success p-2 text-dark bg-opacity-25">25% opacity </div>
<div class="bg-success p-2 text-dark bg-opacity-10">10% opacity </div>

```



TABLE:

1. Basic Table:

- The **.table** class adds basic styling (light padding and horizontal dividers) to a table.

Example:

```
<table class="table" >
  <thead>
    <tr>
      <th>Firstname</th>
      <th>Lastname</th>
      <th>Email</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>John</td>
      <td>Doe</td>
      <td>john@example.com</td>
    </tr>
    <tr>
      <td>Mary</td>
      <td>Moe</td>
      <td>mary@example.com</td>
    </tr>
  </tbody>
</table>
```

2. Striped Rows:

- The **.table-striped** class adds zebra-stripes to a table.
- Example:

```
<table class="table table-striped">
```

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

3. Bordered Table:

- The **.table-bordered** class adds borders on all sides of the table and cells.
- Example:

```
<table class="table table-bordered">
```

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

4. Hover Rows:

- The **.table-hover** class adds a hover effect (grey background color) on table rows.
- Example:

```
<table class="table table-hover">
```

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

5. Black/Dark table:

- The **.table-dark** class adds a black background to the table.
- Example:

```
<table class="table table-dark">
```

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

6. Dark Striped table:

- Combine .table-dark and .table-striped to create a dark, striped table.
- Example:

```
<table class="table table-dark table-striped">
```

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

7. Hoverable Dark table:

- The .table-hover class adds a hover effect (grey background color) on table rows.
- Example:

```
<table class="table table-dark table-hover">
```

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

8. Borderless Table:

- The .table-borderless class removes borders from the table.
- Example:

```
<table class="table table-borderless">
```

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

9. Contextual table:

- Contextual classes can be used to color the whole table (<table>), the table rows (<tr>) or table cells (<td>).
- Example:

```
<div class="container mt-3">
```

```
<table class="table">
```

```
<thead>
```

```
<tr>
```

```
<th>Firstname</th>
```

```
<th>Lastname</th>
```

```
<th>Email</th>
```

```
</tr>
</thead>
<tbody>
  <tr>
    <td>Default</td>
    <td>Defaultson</td>
    <td>def@somemail.com</td>
  </tr>
  <tr class="table-primary">
    <td>Primary</td>
    <td>Joe</td>
    <td>joe@example.com</td>
  </tr>
  <tr class="table-success">
    <td>Success</td>
    <td>Doe</td>
    <td>john@example.com</td>
  </tr>
  <tr class="table-danger">
    <td>Danger</td>
    <td>Moe</td>
    <td>mary@example.com</td>
  </tr>
  <tr class="table-info">
    <td>Info</td>
    <td>july@example.com</td>
  </tr>
  <tr class="table-warning">
    <td>Warning</td>
    <td>Refs</td>
    <td>bo@example.com</td>
  </tr>
  <tr class="table-active">
    <td>Active</td>
    <td>Activeson</td>
    <td>act@example.com</td>
  </tr>
```

```

<tr class="table-secondary">
  <td>Secondary</td>
  <td>Secondson</td>
  <td>sec@example.com</td>
</tr>
<tr class="table-light">
  <td>Light</td>
  <td>Angie</td>
  <td>angie@example.com</td>
</tr>
<tr class="table-dark">
  <td>Dark</td>
  <td>Bo</td>
  <td>bo@example.com</td>
</tr>
</tbody>
</table>
</div>

```

Firstname	Lastname	Email
Default	Defaultson	def@somemail.com
Primary	Joe	joe@example.com
Success	Doe	john@example.com
Danger	Moe	mary@example.com
Info	Dooley	july@example.com
Warning	Refs	bo@example.com
Active	Activeson	act@example.com
Secondary	Secondson	sec@example.com
Light	Angie	angie@example.com
Dark	Bo	bo@example.com

10. Table Head Colors:

- You can also use any of the contextual classes to only add a background color to the table header.
- Example:

```
<thead class="table-dark">
```

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

11. Small Table:

- The `.table-sm` class makes the table smaller by cutting cell padding in half.
- Example:

```
<table class="table table-bordered table-sm">
```

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

12. Responsive Table:

- The `.table-responsive` class adds a scrollbar to the table when needed (when it is too big horizontally).
- Example:

```
<div class="table-responsive">
```

Country	Sex	Example	Example	Example	Example	Exam
USA	Female	Yes	Yes	Yes	Yes	Yes




IMAGE:

1. Rounded image:

- The `.rounded` class adds rounded corners to an image.
- Example:

```

```



2. Circled Image:

- The `.rounded-circle` class shapes the image to a circle.
- Example:

``



3. Bordered Image:

- The **.img-thumbnail** class shapes the image to a thumbnail (bordered).
- Example:

``



4. Left align:

- Float an image to the left with the **.float-start** class.
- Example:

``

5. Right Align:

- Float an image to the right with **.float-end** class.
- Example:

``

6. Center align:

- Center an image by adding the utility classes **.mx-auto** (margin:auto) and **.d-block** (display:block) to the image.
- Example:

``

7. Responsive image:

- Responsive images automatically adjust to fit the size of the screen.
- Create responsive images by adding an **.img-fluid** class to the `` tag. The image will then scale nicely to the parent element.

- The **.img-fluid** class applies max-width: 100%; and height: auto; to the image.
- Example:

```

```

DISPLAY:

- none
- inline
- inline-block
- block
- grid
- inline-grid
- table
- table-cell
- table-row
- flex
- inline-flex

Example:

```
<div class="d-inline p-2 text-bg-primary">d-inline</div>
```

```
<div class="d-inline p-2 text-bg-dark">d-inline</div>
```

```
<span class="d-block p-2 text-bg-primary">d-block</span>
```

```
<span class="d-block p-2 text-bg-dark">d-block</span>
```



GAP:

- When using display: grid, you can make use of gap utilities on the parent grid container.

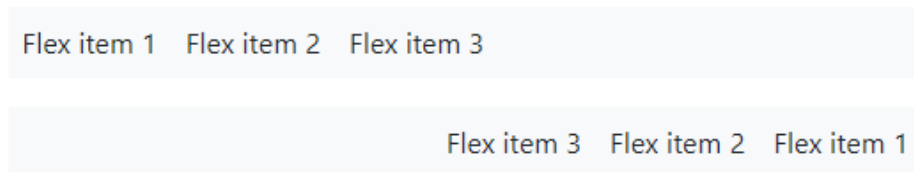
➤ **Example:**

```
<div class="d-grid gap-3">  
  <div class="p-2 bg-light border">Grid item 1</div>  
  <div class="p-2 bg-light border">Grid item 2</div>  
  <div class="p-2 bg-light border">Grid item 3</div>  
</div>
```



FLEX-DIRECTION:

- Use `.flex-row` to set a horizontal direction (the browser default), or `.flex-row-reverse` to start the horizontal direction from the opposite side.



Example:

```
<div class="d-flex flex-row mb-3">  
  <div class="p-2">Flex item 1</div>  
  <div class="p-2">Flex item 2</div>  
  <div class="p-2">Flex item 3</div>  
</div>  
  
<div class="d-flex flex-row-reverse">  
  <div class="p-2">Flex item 1</div>
```

```

<div class="p-2">Flex item 2</div>

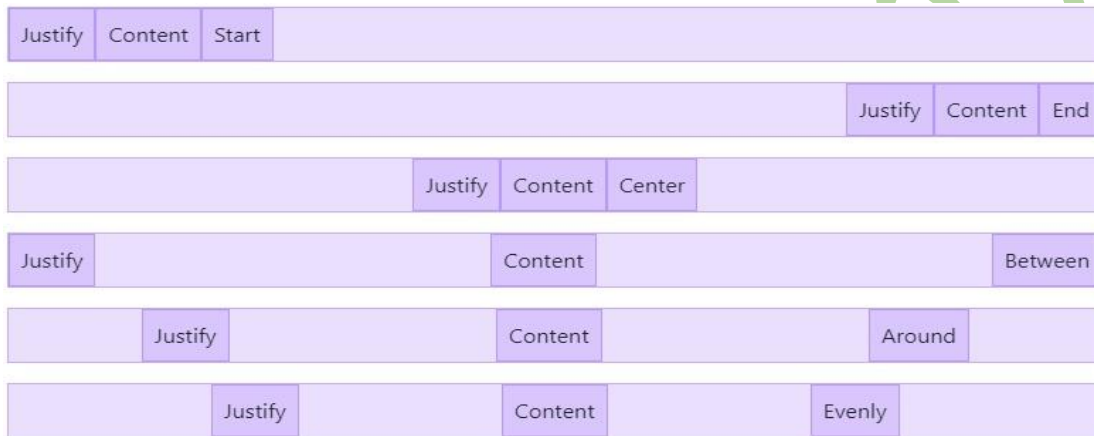
<div class="p-2">Flex item 3</div>

</div>

```

JUSTIFY CONTENT:

- Use justify-content utilities on flexbox containers to change the alignment of flex items on the main axis (the x-axis to start, y-axis if flex-direction: column). Choose from **start** (browser default), **end**, **center**, **between**, **around**, or **evenly**.



- **Example:**

```

<div class="d-flex justify-content-start">...</div>

<div class="d-flex justify-content-end">...</div>

<div class="d-flex justify-content-center">...</div>

<div class="d-flex justify-content-between">...</div>

<div class="d-flex justify-content-around">...</div>

<div class="d-flex justify-content-evenly">...</div>

```

GROW AND SHRINK:

1. The .flex-grow-1 elements uses all available space it can, while allowing the remaining two flex items their necessary space.

```

<div class="d-flex">

  <div class="p-2 flex-grow-1">Flex item</div>

  <div class="p-2">Flex item</div>

```



```
<div class="p-2">Third flex item</div>
</div>
```

Flex item	Flex item	Third flex item
-----------	-----------	-----------------

- The second flex item with **.flex-shrink-1** is forced to wrap its contents to a new line, “shrinking” to allow more space for the previous flex item with **.w-100**

Example:

```
<div class="d-flex">
  <div class="p-2 w-100">Flex item</div>
  <div class="p-2 flex-shrink-1">Flex item</div>
</div>
```

Flex item	Flex item
-----------	-----------

ORDER:

- We only provide options for making an item first or last, as well as a reset to use the DOM order. As order takes any integer value from 0 to 5, add custom CSS for any additional values needed.

➤ Example:

```
<div class="d-flex flex-nowrap">
  <div class="order-3 p-2">First flex item</div>
  <div class="order-2 p-2">Second flex item</div>
  <div class="order-1 p-2">Third flex item</div>
</div>
```

Third flex item	Second flex item	First flex item
-----------------	------------------	-----------------

FLOAT:

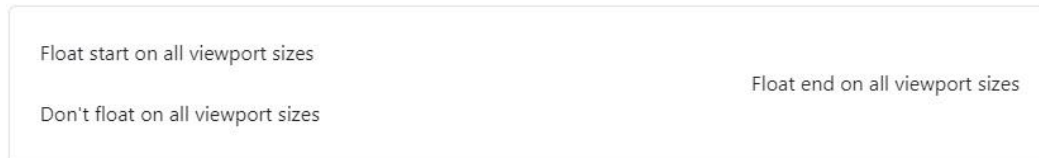
- These utility classes float an element to the left or right, or disable floating.

➤ Example:

```
<div class="float-start">Float start </div><br>
```

```
<div class="float-end">Float end </div><br>
```

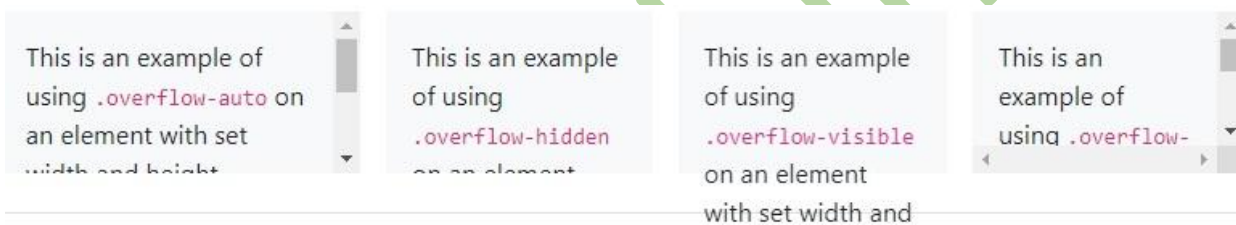
```
<div class="float-none">Don't float on all viewport </div>
```



OVERFLOW:

```
<div class="overflow-auto">...</div>
```

```
<div class="overflow-hidden">...</div>
```



```
<div class="overflow-visible">...</div>
```

```
<div class="overflow-scroll">...</div>
```

POSITION:

```
<div class="position-static">...</div>
```

```
<div class="position-relative">...</div>
```

```
<div class="position-absolute">...</div>
```

```
<div class="position-fixed">...</div>
```

```
<div class="position-sticky">...</div>
```

ARRANGE ELEMENTS:

➤ Arrange elements easily with the edge positioning utilities. The format is {property}-{position}.

➤ Where *property* is one of:

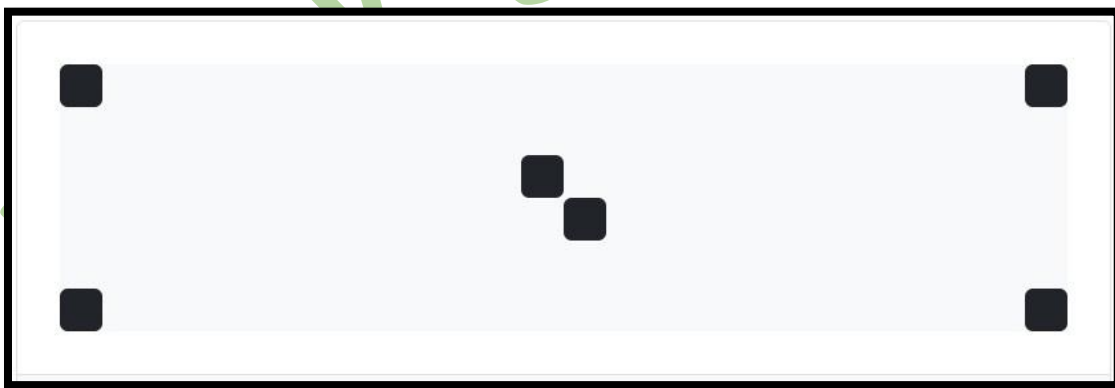
- top - for the vertical top position
- start - for the horizontal left position (in LTR)
- bottom - for the vertical bottom position
- end - for the horizontal right position (in LTR)

➤ Where *position* is one of:

- 0 - for 0 edge position
- 50 - for 50% edge position
- 100 - for 100% edge position

➤ Example:

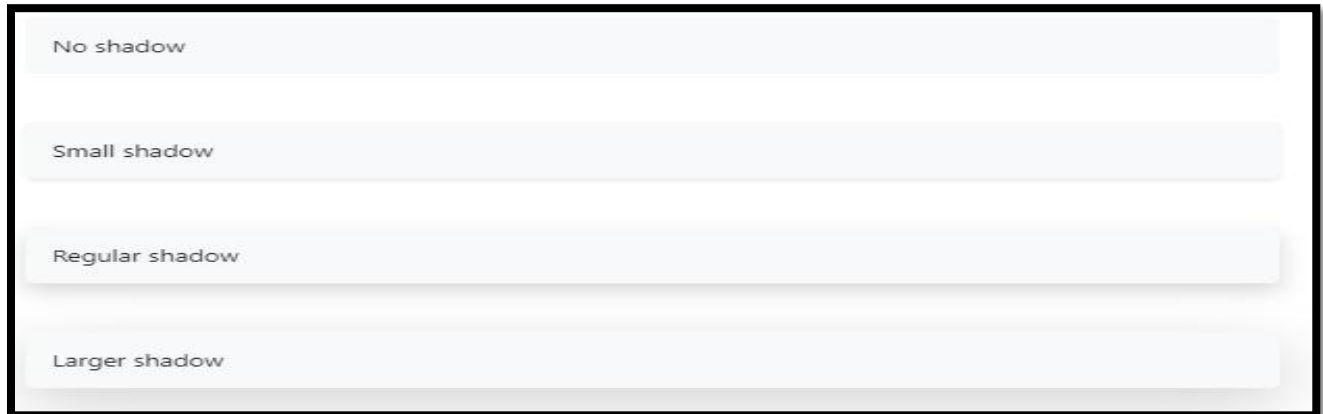
```
<div class="position-absolute top-0 start-0"></div>
<div class="position-absolute top-0 end-0"></div>
<div class="position-absolute top-50 start-50"></div>
<div class="position-absolute bottom-50 end-50"></div>
<div class="position-absolute bottom-0 start-0"></div>
<div class="position-absolute bottom-0 end-0"></div>
```



SHADOWS:

```
<div class="shadow-none p-3 mb-5 bg-body-tertiary rounded">No shadow</div>
<div class="shadow-sm p-3 mb-5 bg-body-tertiary rounded">Small shadow</div>
<div class="shadow p-3 mb-5 bg-body-tertiary rounded">Regular shadow</div>
```

`<div class="shadow-lg p-3 mb-5 bg-body-tertiary rounded">Larger shadow</div>`



SIZING(WIDTH):

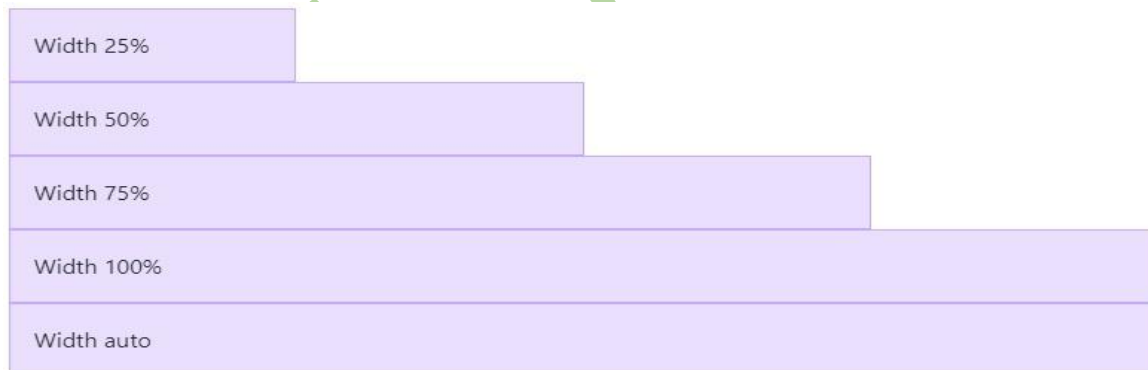
`<div class="w-25 p-3">Width 25%</div>`

`<div class="w-50 p-3">Width 50%</div>`

`<div class="w-75 p-3">Width 75%</div>`

`<div class="w-100 p-3">Width 100%</div>`

`<div class="w-auto p-3">Width auto</div>`



SIZING(HEIGHT):

`<div style="height: 100px;">`

`<div class="h-25 d-inline-block" style="width: 120px;">Height 25%</div>`

`<div class="h-50 d-inline-block" style="width: 120px;">Height 50%</div>`

`<div class="h-75 d-inline-block" style="width: 120px;">Height 75%</div>`

`<div class="h-100 d-inline-block" style="width: 120px;">Height 100%</div>`

```
<div class="h-auto d-inline-block" style="width: 120px;">Height auto</div>  
</div>
```



TEXT WRAPPING AND OVERFLOW:

- Wrap text with a .text-wrap class.

- Example1:

```
<div class="badge bg-primary text-wrap"  
style="width:6rem;">  
  This text should wrap.  
</div>
```

This text
should wrap.

- Wrap text with a .text-nowrap class.

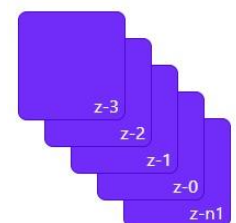
- Example2:

```
<div class="text-nowrap bg-body-  
secondary border" style="width: 8rem;">  
  This text should overflow the parent.  
</div>
```

This text should overflow the parent.

Z-INDEX:

```
<div class="z-3 position-absolute p-5 rounded-3"><span>z-3</span></div>  
<div class="z-2 position-absolute p-5 rounded-3"><span>z-2</span></div>  
<div class="z-1 position-absolute p-5 rounded-3"><span>z-1</span></div>  
<div class="z-0 position-absolute p-5 rounded-3"><span>z-0</span></div>  
<div class="z-n1 position-absolute p-5 rounded-3"><span>z-n1</span></div>
```

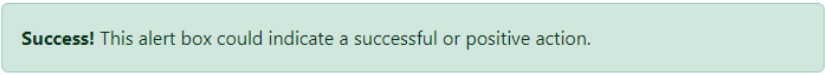


ALERTS:

- Bootstrap 5 provides an easy way to create predefined alert messages.
- Alerts are created with the `.alert` class, followed by one of the contextual classes `.alert-success`, `.alert-info`, `.alert-warning`, `.alert-danger`, `.alert-primary`, `.alert-secondary`, `.alert-light` or `.alert-dark`.

Example1:(Alert)

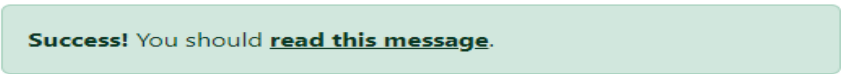
```
<div class="alert alert-success">  
    <strong>Success!</strong> This alert box could indicate a  
    successful or positive action.  
</div>
```



Success! This alert box could indicate a successful or positive action.

Example2:(Alert link)

```
<div class="alert alert-success">  
    <strong>Success!</strong> You should <a href="#" class="alert-  
    link">read this message</a>.  
</div>
```



Success! You should [read this message](#).

Example3:(Closing Alerts)

```
<div class="alert alert-success alert-dismissible">  
    <button type="button" class="btn-close" data-bs-dismiss="alert"></button>  
    <strong>Success!</strong> <This alert box could indicate a successful or  
    positive action.  
</div>
```

Success! This alert box could indicate a successful or positive action.



Example4: (Animated Alerts)

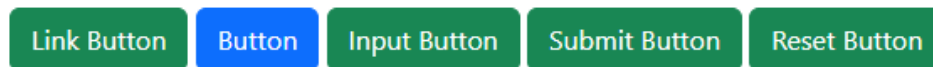
```
<div class="alert alert-light alert-dismissible fade show">  
  <button type="button" class="btn-close" data-bs-dismiss="alert"></button>  
  <strong>Light!</strong> Light grey alert.  
</div>
```

BUTTON:

1. Button Styles:

- Bootstrap 5 provides different styles of buttons.
- The button classes can be used on `<a>`, `<button>`, or `<input>` elements.
- Example:

```
<button type="button" class="btn btn-primary">Primary</button>  
<input type="button" class="btn btn-success" value="Input Button">  
<input type="submit" class="btn btn-success" value="Submit Button">  
<input type="reset" class="btn btn-success" value="Reset Button">
```



2. Button Outline:

- Bootstrap 5 also provides eight outline/bordered buttons.
- Move the mouse over them to see an additional “hover” effect.
- Example:

```
<button type="button" class="btn btn-outline-secondary">Secondary</button>
```



3. Button Sizes:

- Use the `.btn-lg` class for large buttons or `.btn-sm` class for small buttons.
- Example:

```
<button type="button" class="btn btn-primary btn-lg">Large</button>
```

Button Sizes

Large

Default

Small

4. Block level buttons:

- To create a block level button that spans the entire width of the parent element, use the `.d-grid` "helper" class on the parent element.

- Example:

```
<div class="d-grid">  
  <button type="button" class="btn btn-primary btn-block">Full-  
    Width Button</button>  
</div>
```



5. Active/Disable button:

- A button can be set to an active (appear pressed) or a disabled (unclickable) state.
- The class `.active` makes a button appear pressed, and the `disabled` attribute makes a button unclickable. Note that `<a>` elements do not support the disabled attribute and must therefore use the `.disabled` class to make it visually appear disabled.

- Example:

```
<button type="button" class="btn btn-primary disabled">Disabled  
Primary</button>
```



6. Spinner Buttons:

- Example:

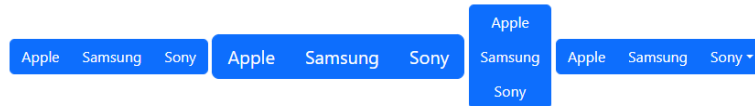
```
<button class="btn btn-primary disabled">  
  <span class="spinner-grow spinner-grow-sm"></span>  
  Loading..  
</button>
```



7. Grouped button:

- Bootstrap 5 allows you to group a series of buttons together(on a single line) in a button group.

- Use a `<div>` element with class **.btn-group** to create a button group.



1. Example:

```
<div class="btn-group">
  <button type="button" class="btn btn-
primary">Apple</button>
  <button type="button" class="btn btn-
primary">Samsung</button>
  <button type="button" class="btn btn-primary">Sony</button>
</div>
```



BADGES:

- Badges are used to add additional information to any content.
- Use the **.badge** class together with a contextual class (like **.bg-secondary**) within `` elements to create rectangular badges.

1. Contextual Badges.

```
<span class="badge bg-info">Info</span>
```

2. Pill Badges.

```
<span class="badge rounded-pill bg-info">Info</span>
```

3. Badge inside an Element.

```
<button type="button" class="btn btn-info">Message<span
class="badge bg-danger">4</span></button>
```

SPINNERS:

1. Simple Spinners:

- To create a spinner/loader, use the **.spinner-border** class.
- Example:

```
<div class="spinner-border"></div>
```

2. Colored Spinners:

- Use any text color utilities to add a color to the spinner.
- Example:

```
<div class="spinner-border text-primary"></div>
```

3. Growing Spinners:

- Use the **.spinner-grow** class if you want the spinner/loader to grow instead of "spin".



- Example:

```
<div class="spinner-grow text-info"></div>
```

4. Spinners Size:

- Use `.spinner-border-sm` or `.spinner-grow-sm` to create a smaller spinner.

- Example:

```
<div class="spinner-border spinner-border-sm"></div>
```

5. Spinner Buttons:

- You can also add spinners to a button, with or without text.

- Example:

```
<button class="btn btn-primary">  
  <span class="spinner-border spinner-border-sm"></span>  
</button>
```

CARDS:

- A card in Bootstrap 5 is a bordered box with some padding around its content. It includes options for headers, footers, content, colors, etc.

1. Basic card:

- ✓ A basic card is created with the `.card` class, and content inside the card has a `.card-body` class.

- ✓ Example:

```
<div class="card">  
  <div class="card-body">Basic card</div>  
</div>
```

2. Header and Footer:

- ✓ The `.card-header` class adds a heading to the card and the `.card-footer` class adds a footer to the card.

- ✓ Example:

```
<div class="card">  
  <div class="card-header">Header</div>  
  <div class="card-body">Content</div>  
  <div class="card-footer">Footer</div>  
</div>
```

3. Contextual cards:

- ✓ To add a background color the card, use contextual classes (`.bg-primary`, `.bg-success`, `.bg-info`, `.bg-warning`, `.bg-danger`, `.bg-secondary`, `.bg-dark` and `.bg-light`).

- ✓ Example:

```
<div class="card bg-primary text-white">
  <div class="card-body">Primary card</div>
</div>
```

4. Title, text and links:

- ✓ Use **.card-title** to add card titles to any heading element. The **.card-text** class is used to remove bottom margins for a `<p>` element if it is the last child (or the only one) inside **.card-body**. The **.card-link** class adds a blue color to any link, and a hover effect.

- ✓ Example:

```
<div class="card">
  <div class="card-body">
    <h4 class="card-title">Card title</h4>
    <p class="card-text">Some example text. Some
      example text.</p>
    <a href="#" class="card-link">Card link</a>
    <a href="#" class="card-link">Another link</a>
  </div>
</div>
```

5. Card images:

- ✓ Add **.card-img-top** or **.card-img-bottom** to an `` to place the image at the top or at the bottom inside the card. Note that we have added the image outside of the **.card-body** to span the entire width.

- ✓ Example:

```
<div class="card" style="width:400px">
  
  <div class="card-body">
    <h4 class="card-title">John Doe</h4>
    <p class="card-text">Some example text.</p>
    <a href="#" class="btn btn-primary">See
      Profile</a>
  </div>
</div>
```

6. Card image overlays:

- ✓ Turn an image into a card background and use **.card-img-overlay** to add text on top of the image.

- ✓ Example:

```

<div class="card" style="width:500px">
  
  <div class="card-img-overlay">
    <h4 class="card-title">John Doe</h4>
    <p class="card-text">Some example text.</p>
    <a href="#" class="btn btn-primary">See
    Profile</a>
  </div>
</div>

```

CAROUSEL / SLIDESHOW:

- The Carousel is a slideshow for cycling through elements.
- Example:

```

<!-- Carousel -->
<div id="demo" class="carousel slide" data-bs-ride="carousel">

```

```

  <!-- Indicators/dots -->
  <div class="carousel-indicators">
    <button type="button" data-bs-target="#demo" data-bs-slide-to="0"
    class="active"></button>
    <button type="button" data-bs-target="#demo" data-bs-slide-
    to="1"></button>
    <button type="button" data-bs-target="#demo" data-bs-slide-
    to="2"></button>
  </div>

```

```

  <!-- The slideshow/carousel -->
  <div class="carousel-inner">
    <div class="carousel-item active">
      
    </div>
    <div class="carousel-item">
      
    </div>
    <div class="carousel-item">

```

```

        
    </div>
</div>

```

```

        <!-- Left and right controls/icons -->
        <button class="carousel-control-prev" type="button" data-bs-
target="#demo" data-bs-slide="prev">
            <span class="carousel-control-prev-icon"></span>
        </button>
        <button class="carousel-control-next" type="button" data-bs-
target="#demo" data-bs-slide="next">
            <span class="carousel-control-next-icon"></span>
        </button>
    </div>

```

Explanation:

Class	Description
.carousel	Creates a carousel
.carousel-indicators	Adds indicators for the carousel. These are the little dots at the bottom of each slide (which indicates how many slides there are in the carousel, and which slide the user are currently viewing)
.carousel-inner	Adds slides to the carousel
.carousel-item	Specifies the content of each slide
.carousel-control-prev	Adds a left (previous) button to the carousel, which allows the user to go back between the slides
.carousel-control-	Adds a right (next) button to the carousel, which allows the

next	user to go forward between the slides
.carousel-control-prev-icon	Used together with .carousel-control-prev to create a "previous" button
.carousel-control-next-icon	Used together with .carousel-control-next to create a "next" button
.slide	Adds a CSS transition and animation effect when sliding from one item to the next. Remove this class if you do not want this effect

NAVIGATION BARS:

- With Bootstrap, a navigation bar can extend or collapse, depending on the screen size.
- A standard navigation bar is created with the **.navbar** class, followed by a responsive collapsing class: **.navbar-expand-xxl|xl|lg|md|sm** (stacks the navbar vertically on xxlarge, extra large, large, medium or small screens).
- To add links inside the navbar, use either an element (or a <div>) with **class="navbar-nav"**. Then add elements with a .nav-item class followed by an <a> element with a **.nav-link** class.