FULLSTACK BOOTSTRAP

INDEX: -

Introduction

- What
- Why

Basic

- Default setting
- Cotainer
- Sizing
- Float
- Spacing
- Border
- Colors
- Text size
- Text design
- Other typography

Grids

- Equal columns
- Responsive columns
- Grid options
- Row cols
- Gutters
- Order
- Offset
- Grid gap

CSS

- Image
- Table
- Background
- Font
- Text
- Link
- Interaction
- Media setup
- Element design

- Position
- Display
- Focus ring
- Stretched links
- Flex
- Flex uses
- Icons

Components 1

- Buttons
- Button group
- Collapse
- Navs and tabs
- Dropdowns

Forms

- Form basic
- Disabled and readonly
- Input
- Datalist
- Select
- Checkbox and radio
- Range
- Input group
- Floating labels
- Form layout

Components 2

- Breadcrumb
- Badges
- List group
- Cards
- Close button
- Offcanvas
- Navbar
- Placeholders
- Progress
- Spinners
- Accordion
- Pagination
- Carousel
- Scrollspy

Themes

- Download and apply
- Customize theme
- Customize class properties

Javascript

- checkbox
- tooltips
- Form validation
- Alerts
- Scrollspy
- Popovers
- Toasts
- Modal

Introduction to Bootstrap: -

What: -

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

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

Bootstrap versions: -

Bootstrap 5 (released 2021) is the newest version of Bootstrap (released 2013); with new components, faster stylesheet and more responsiveness.

Bootstrap 5 supports the latest, stable releases of all major browsers and platforms. However, Internet Explorer 11 and down is not supported.

The main differences between Bootstrap 5 and Bootstrap 3 & 4, is that Bootstrap 5 has switched to vanilla JavaScript instead of jQuery.

Why: -

- Easy to use: Anybody with just basic knowledge of HTML and CSS can start using Bootstrap
- Responsive features: Bootstrap's responsive CSS adjusts to phones, tablets, and desktops
- Mobile-first approach: In Bootstrap, mobile-first styles are part of the core framework
- Browser compatibility: Bootstrap 5 is compatible with all modern browsers (Chrome, Firefox, Edge, Safari, and Opera).

Installation: -

Download bootstrap: -

- Go to https://getbootstrap.com/ website and click on download
- Then download 'Compiled CSS and JS file' and extract it in folder
- Then copy file path of min CSS file and paste it in link's 'href' attributes for link CSS file.
- Then copy file path of min JS file and paste it in script's 'src' attributes for link JS file

Ex: -

```
<link href="C:\Users\IANT\Downloads\bootstrap-5.3.3-dist\css\bootstrap.min.css" media="all"
rel="stylesheet">
<script src="C:/Users/IANT/Downloads/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"
integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz"
crossorigin="anonymous"></script>
```

CDN (Content Delivery Network) link: -

- Go to https://getbootstrap.com/ website and click on download
- In CDN via jsDelivr copy CDN links for CSS and JS, then paste them in html file.
- Paste CSS link tag in head tag of html.
- Paste JS Script tag at end of body tag of html.

CSS file link: -

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH"
crossorigin="anonymous">
```

JS file link: -

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"
integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz"
crossorigin="anonymous"></script>
```

Popper: - it will be include if using compiled JS for dependencies. It link will be add before JS file link

Popper link: -

```
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.8/dist/umd/popper.min.js"
integrity="sha384-I7E8VVD/ismYTF4hNIPjVp/Zjvgyol6VFvRkX/vR+Vc4jQkC+hVqc2pM8ODewa9r"
crossorigin="anonymous"></script>
```

Js file link: -

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.min.js"
integrity="sha384-0pUGZvbkm6XF6gxjEnlmuGrJXVbNuzT9qBBavbLwCsOGabYfZo0T0to5eqruptLy"
crossorigin="anonymous"></script>
```

Bootstrap Basics: -

Bootstrap default setting: - 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).

Container: - Containers are used to pad the content inside of them, and there are two container classes available

Fixed container: - this class provides a responsive fixed width container. This class’s max-width will change on different screen size (breakpoint).

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

class="container"

Ex: - <div class="container text-bg-primary"> Good day </div>

Fluid container: - this class provides a full width container, spanning the entire width of the viewport. This class's width is always 100%

class="fluid-container"
Ex: - <div class="fluid-container text-bg-success"> Fluid container </div>

Container breakpoints: - there are different container classes in bootstrap, for determine when container will be responsive

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

Ex: -

```
<div class="container-sm border"> Container-sm </div>
<div class="container-md border"> Container-md </div>
<div class="container-lg border"> Container-lg </div>
<div class="container-xl border"> Container-xl </div>
<div class="container-xxl border"> Container-xxl </div>
```

Sizing: -

Relative to the parent: -

Width: -

```
<div class="border w-25">width 25%</div>
<div class="border w-50">width 50%</div>
<div class="border w-75">width 75%</div>
<div class="border w-100">width 100%</div>
<div class="border w-auto">width auto</div>
```

Height: - height class will show effect if it contain in fixed height container

```
<div style="height: 100px; border:1px solid black;">
  <div class="border h-25">height 25%</div>
</div>
<div style="height: 100px; border:1px solid black;">
  <div class="border h-50">height 50%</div>
</div>
<div style="height: 100px; border:1px solid black;">
  <div class="border h-75">height 75%</div>
</div>
<div style="height: 100px; border:1px solid black;">
```

```
    <div class="border h-100">height 100%</div>
</div>
<div style="height: 100px; border:1px solid black;">
    <div class="border h-auto">height auto</div>
</div>
```

Float: -

Float start: -

class="float-start"

Ex: - <div class="border float-start">float start</div>

Float start with breakpoints: - apply float start when viewport size is equal or wider than breakpoints

1) float-sm-start
2) float-md-start
3) float-lg-start
4) float-xl-start
5) float-xxl-start

Float end: -

class="float-end"

Ex: - <div class="border float-end ">float end </div>

Float end with breakpoints: - apply float end when viewport size is equal or wider than breakpoints

1) float-sm-end
2) float-md-end
3) float-lg-end
4) float-xl-end
5) float-xxl-end

Float none: -

class="float-none"

Ex: - <div class="border float-none ">float none </div>

Float none with breakpoints: - apply float none when viewport size is equal or wider than breakpoints

1) float-sm-none
2) float-md-none
3) float-lg-none
4) float-xl-none
5) float-xxl-none

Clearfix: - Quickly and easily clear floated content within a container by adding a clearfix utility.

Ex: -

```
<div class="border clearfix border-dark">
    <div class="float-start">good day</div>
</div>
<div class="border">hello day</div>
```

Spacing: -

Margin: -

Margin with size: -

```
<div class="border border-dark m-0">m-0</div>
<div class="border border-dark m-1">m-1</div>
<div class="border border-dark m-2">m-2</div>
<div class="border border-dark m-3">m-3</div>
<div class="border border-dark m-4">m-4</div>
<div class="border border-dark m-5">m-5</div>
```

Margin particular side: -

```
<div class="border border-dark mt-5">mt-5</div> //top
<div class="border border-dark mb-5">mb-5</div> //bottom
<div class="border border-dark ms-5">ms-5</div> //start
<div class="border border-dark me-5">me-5</div> //end
<div class="border border-dark mx-5">mx-5</div> //start & end
<div class="border border-dark my-5">my-5</div> //top & bottom
```

Margin with breakpoint: -

```
<div class="border border-dark my-5 mx-sm-3 mx-xl-5"> my-5 mx-sm-3 mx-xl-5 </div> //margin-breakpoint-size
```

Padding: -

Padding with size: -

```
<div class="border border-dark p-0"> p-0</div>
<div class="border border-dark p-1"> p-1</div>
<div class="border border-dark p-2"> p-2</div>
<div class="border border-dark p-3"> p-3</div>
<div class="border border-dark p-4"> p-4</div>
<div class="border border-dark p-5"> p-5</div>
```

Padding particular side: -

```
<div class="border border-dark pt-5"> pt-5</div> //top
<div class="border border-dark pb-5"> pb-5</div> //bottom
<div class="border border-dark ps-5"> ps-5</div> //start
<div class="border border-dark pe-5"> pe-5</div> //end
<div class="border border-dark px-5"> px-5</div> //start & end
<div class="border border-dark py-5"> py-5</div> //top & bottom
```

Padding with breakpoint: -

```
<div class="border border-dark py-5 px-sm-3 px-xl-5"> py-5 px-sm-3 px-xl-5 </div> //padding-breakpoint-size
```

Border: - set border using bootstrap

Border: - class="border"

Ex: - <div class="border"> good </div>

Additive borders: -

Border at single side: - class="container border-top"

Ex: -

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

Multiple side borders: -

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

Subtractive border: -

Remove border: - class="border border-top-0"

Ex: -

```
<div class="container border border-top-0"> subtractive border top </div>
<div class="container border border-start-0"> subtractive border start </div>
<div class="container border border-end-0"> subtractive border end </div>
<div class="container border border-bottom-0"> subtractive border bottom </div>
```

Subtractive multiple borders: -

```
<div class="container border border-start-0 border-end-0"> subtractive border start end </div>
```

Border colors: -

Ex: -

```
<div class="container border border-primary">border primary</div>
<div class="container border border-secondary">border secondary</div>
<div class="container border border-success">border success</div>
<div class="container border border-danger">border danger</div>
<div class="container border border-warning">border danger</div>
<div class="container border border-info">border info</div>
<div class="container border border-light">border light</div>
<div class="container border border-dark">border dark</div>
<div class="container border border-primary-subtle">border primary subtle</div>
```

Use subtle with other border-color classes also.

Border opacity: -

Ex: -

```
<div class="container border border-dark border-opacity-75">border opacity 75%</div>
```

```
<div class="container border border-dark border-opacity-50">border opacity 50%</div>
<div class="container border border-dark border-opacity-25">border opacity 25%</div>
<div class="container border border-dark border-opacity-10">border opacity 10%</div>
```

Border width: -

Ex: -

```
<div class="container border border-1">border 1 width</div>
<div class="container border border-2">border 2 width</div>
<div class="container border border-3">border 3 width</div>
<div class="container border border-4">border 4 width</div>
<div class="container border border-5">border 5 width</div>
```

Border radius: -

```
<div class="container border rounded">border rounded</div>
<div class="container border rounded-top">border rounded top</div>
<div class="container border rounded-end">border rounded-end</div>
<div class="container border rounded-start">border rounded-start</div>
<div class="container border rounded-bottom">border rounded-bottom</div>
<div class="container border rounded-pill">border rounded-pill</div>
<div class="container border rounded-circle" style="width: 100px; height: 100px;">border rounded-circle</div>
<div class="container border rounded-start-circle" style="width: 100px; height: 100px;">border rounded start circle</div> //apply circle to start only
```

Border radius sizes: -

```
<div class="container border rounded-0">border rounded 0 size</div>
<div class="container border rounded-1">border rounded 1 size</div>
<div class="container border rounded-2">border rounded 2 size</div>
<div class="container border rounded-3">border rounded 3 size</div>
<div class="container border rounded-4">border rounded 4 size</div>
<div class="container border rounded-5">border rounded 5 size</div>
<div class="container border rounded-start-5">border rounded bottom 5</div> //apply radius size to single size
<div class="container border rounded-5 rounded-top-0">border rounded 5 rounded top 0</div> //remove radius from single side
```

Colors: -

```
<style>
.dback{
   width: 50%;
   height: 40px;
}
</style>
```

Background colors: - set classes for background colors

Primary: - <div class="dback bg-primary"></div>
Success: - <div class="dback bg-success"></div>
Info: - <div class="dback bg-info"></div>
Warning: - <div class="dback bg-warning"></div>
Danger: - <div class="dback bg-danger"></div>
Secondary: - <div class="dback bg-secondary"></div>
Dark: - <div class="dback bg-dark"></div>
Light: - <div class="dback bg-light"></div>
Subtle: - <div class="dback bg-primary-subtle"></div> //also add subtle will other background classes
black: - <div class="dback bg-black"></div>

Text colors: - Bootstrap 5 has some contextual classes that can be used to provide "meaning through colors".

Text muted: - <p class="text-muted">this text is muted</p>
Text primary: - <p class="text-primary">this text is important</p>
Text success: - <p class="text-success">this text indicates success</p>
Text info: - <p class="text-info">this text represent information</p>
Text warning: - <p class="text-warning">this text represent warning</p>
Text danger: - <p class="text-danger">this text represent danger</p>
Text secondary: - <p class="text-secondary">secondary text</p>
Text dark: - <p class="text-dark">dark text</p>
Text body: - (default text color) <p class="text-body">body text</p>
Style: -

```
.bgblack{
   background-color: black;
}
```

Text white: - <p class="text-white bgblack">white text</p>
Text light (#F8F9FA): - <p class="text-light bgblack">light text</p>
Opacity 50% for black and white text: -
<p class="text-black-50">Opacity black text</p>
<p class="text-white-50 bgblack">Opacity white text</p>

Text background colors: - bg-color classes did not work well with text-color classes, text-bg-color classes will automatically handle the appropriate text color for each background color.

Primary: - <div class="dback text-bg-primary">Important</div>
Success: - <div class="dback text-bg-success">success</div>
Info: - div class="dback text-bg-info">information</div>
Warning : - <div class="dback text-bg-warning">warning</div>

Danger: - <div class="dback text-bg-danger">danger</div>
Secondary: - <div class="dback text-bg-secondary">secondary</div>
Dark: - <div class="dback text-bg-dark">dark</div>
Light: - <div class="dback text-bg-light">light</div>

Text size: -

H1 – h6: - HTML headings (<h1> to <h6>) with a bolder font-weight and a responsive font-size.
class="h1" // h1, h2, h3, h4, h5, h6
Ex: - <span class="h1">good day</span>

Display headings: - these are used to stand out more than normal headings (larger font-size and lighter font-weight), and there are six classes to choose from: .display-1 to .display-6:
class="display-1" // display-1, display-2, display-3, display-4, display-5, display-6
Ex: - <p class="display-1">good day</p>

Small: - HTML <small> element (and the .small class) is used to create a smaller, secondary text in any heading:
class= "small"
Ex: - <p class="h1">good day<span class="small">good day</span></p>

Text design: -

Mark: - Bootstrap 5 will style <mark> and .mark with a light background color and some padding
class= "mark"
Ex: - <h1>GOOD Day<span class="mark">GOOD Day</span></h1>

Blockquotes: - Add the .blockquote class to a <blockquote> when quoting blocks of content from another source. And when naming a source, use the .blockquote-footer
class="blockquote"
class="blockquote-footer"
Ex: - <p class="blockquote">good day to all <span class="blockquote-footer">Author</span></p>

Description tags: - Bootstrap will style description elements
<dl> (description list)
<dt> (description title) : - style text bold and height 24px
<dd> (description data) : - style height 24px
Ex: -

```
<dl>
   <dt>GD</dt>
   <dd>good day</dd>
   <dt>GM</dt>
   <dd>good morning</dd>
</dl>
```

Kdb element: - bootstrap will style kbd element with background and text color with padding.
<kbd>: - this element define keyboard input
Ex: - press <kbd>ctrl + p</kbd> for print

Code element: - bootstrap will style kbd element with text-color
<code>: - this element define code in website
Ex: - There are tags named <code>html,head,body</code>

Other typography classes: -

Lead: - Makes a paragraph stand out
Ex: - <p class="lead">good day to all here good bye good good bye</p>

Text start: - Indicates left-aligned text
Ex: - <p class="text-start">good day to all</p>

Text break: - Prevents long text from breaking layout
Ex: -

```
.dt{
   width: 300px;
   border: 1px solid black;
}
<div class="dt">
   <p class="text-break">oifgfighidopcvhuifheoiehpogjgsdsdogjsdpgsfgkldsgklsdr7gvduihnvfpsh</p>
</div>
```

Text center: - Indicates center-aligned text
Ex: - <p class="text-center">good day to all</p>

Text decoration none: - Removes the underline from a link
Ex: - <a href="#" class="text-decoration-none">Good day</a>

Text end: - Indicates right-aligned text
Ex: - <p class="text-end">good day to all</p>

Text nowrap: - Indicates no wrap text
Ex: -

```
.dt{
   width: 300px;
   border: 1px solid black;
}
<div class="dt">
```

```
    <p class="text-nowrap">oifg fighi dopcvh uifheo iehp ogjg sdsdo gjs dpg sfg kldsg klsdr 7gvdu </p>
</div>
```

Text Breakpoints: - also can add breakpoints with text class.

Ex: -

```
<div class="text-sm-break">text-sm-break</div>
<div class="text-md-center text-sm-start">text-md-center text-sm-start</div>
```

Text lowercase: - Indicates lowercased text

Ex: - <p class="text-lowercase">Good daY tO All</p>

Text uppercase: - Indicates uppercased text

Ex: - <p class="text-uppercase">Good daY tO All</p>

Text capitalize: - Indicates capitalized text

Ex: - <p class="text-capitalize">good day to all</p>

Initialism: - Displays the text inside an <abbr> element in a slightly smaller font size

Ex: - <p><abbr title="Good Day" class="initialism">GD</abbr> to all</p>

Note: - abbr tag is depend on title attribute, this tag show dotted border under text

List unstyled: - Removes the default list-style and left margin on list items

Ex: -

```
<ul class="list-inline">
    <li>good</li>
    <li>day</li>
    <li>to</li>
    <li>all</li>
</ul>
```

Bootstrap Grids: - Bootstrap's grid system is built with flexbox and allows up to 12 columns across the page. Also can group the columns together to create wider column, if do not want to use all 12 columns individually. Grid system is responsive, and the columns will re-arrange automatically depending on screen size. Make sure that sum adds up to 12 or fewer.

Equal columns: - define col class container inside row class container, col containers will take equal width in row. Class col take auto columns from 12 columns

Syntax: -

```
<div class="row">
    <div class="col"> </div>
    … … …
</div>
```

Ex: -

```
<div class="row m-0">
   <div class="col border">col 1</div>
   <div class="col border">col 2</div>
   <div class="col border">col 3</div>
</div>
```

Responsive columns: -

for responsive columns define breakpoints and column count for each container in row. Defined col breakpoint will effect if screen size is greater than that breakpoint.

Syntax: -

```
Class="col-breakpoint-assigned columns numbers"
<div class="row">
   <div class="col-lg-5"> </div>
   … … …
</div>
```

Example: -

```
<div class="row m-0">
   <div class="col-sm-1 col-lg-2 border">col 1</div>
   <div class="col-sm-3 col-lg-2 border">col 2</div>
   <div class="col-sm-4 col-lg-4 border">col 3</div>
   <div class="col-sm-3 col-lg-2 border">col 4</div>
   <div class="col-sm-1 col-lg-2 border">col 5</div>
</div>
```

Grid option: -

Breakpoint	Extra small (<576px)	Small (>=576px)	Medium (>=768px)	Large (>=992px)	Extra Large (>=1200px)	XXL (>=1400px)
Class prefix	.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-	.col-xxl-
Grid behaviour	Horizontal at all times	Collapsed to start, horizontal above breakpoints	Collapsed to start, horizontal above breakpoints	Collapsed to start, horizontal above breakpoints	Collapsed to start, horizontal above breakpoints	Collapsed to start, horizontal above breakpoints
Container width	None (auto)	540px	720px	960px	1140px	1320px
Suitable for	Portrait phones	Landscape phones	Tablets	Laptops	Laptops and Desktops	Laptops and Desktops
# of columns	12	12	12	12	12	12

Gutter width	1.5rem (.75rem on each side of a column)	1.5rem (.75rem on each side of a column)	1.5rem (.75rem on each side of a column)	1.5rem (.75rem on each side of a column)	1.5rem (.75rem on each side of a column)	1.5rem (.75rem on each side of a column)
Nestable	Yes	Yes	Yes	Yes	Yes	Yes
Offsets	Yes	Yes	Yes	Yes	Yes	Yes
Column ordering	Yes	Yes	Yes	Yes	Yes	Yes

Row cols: - it controls how many columns can appear in next to each other, regardless of how many “col’s” elements are in row.

Syntax: -

Row-cols-breakpoint-count of columns
class=" row-cols-sm-2"

Ex: -

```
<div class="row row-cols-1 row-cols-sm-2 row-cols-xl-4">
   <div class="col border">col1</div>
   <div class="col border">col2</div>
   <div class="col border">col3</div>
   <div class="col border">col4</div>
</div>
```

Nested rows: - it can set row inside columns

Ex: -

```
<div class="row m-0">
   <div class="row col border m-0 p-0">
      <div class="col border">col1.1</div>
      <div class="col border">col1.2</div>
   </div>
   <div class="col border">col2</div>
</div>
```

Gutters: - Gutters are the padding between your columns, used to responsively space and align content in the Bootstrap grid system.

Horizontal gutters: - gx class handles horizontal spacing in gutter class, with 0 to 5 and break points

Syntax: - gx-breakpoint-number
class="row gx-sm-4"

Ex: -

```
<div class="row gx-sm-4 m-0">
   <div class="col border">col1</div>
   <div class="col border">col2</div>
   <div class="col border ">col3</div>
</div>
```

Vertical gutters: - gy class handles horizontal spacing in gutter class, with 0 to 5 and break points

Syntax: - gy-breakpoint-number

Ex: -

```
class="row gy-sm-4"
<div class="row gy-sm-4 m-0 row-cols-2">
   <div class="col border text-bg-primary">col1</div>
   <div class="col border text-bg-primary">col2</div>
   <div class="col border text-bg-primary">col3</div>
   <div class="col border text-bg-primary">col4</div>
</div>
```

Vertical and horizontal gutters: - .g-* classes to control the horizontal and vertical grid gutters.

Syntax: - g-breakpoint-number

class="row g-sm-4"

Ex: -

```
<div class="row g-5 row-cols-2">
   <div class="col border text-bg-primary">g col1</div>
   <div class="col border text-bg-primary">g col2</div>
   <div class="col border text-bg-primary">g col3</div>
   <div class="col border text-bg-primary">g col4</div>
</div>
```

No gutters: - The gutters between columns in our predefined grid classes can be removed with .g-0. This removes the negative margins from .row and the horizontal padding from all immediate children columns.

Ex: -

```
<div class="row g-0">
   <div class="col border text-bg-primary">col1</div>
   <div class="col border text-bg-primary">col2</div>
</div>
```

Order: - .order- classes for controlling the visual order of your content. These classes are responsive, so you can set the order by breakpoint, support order 1 to 5 also can use "last" and "first" with order

Syntax: - order-breakpoint-number

class="col order-md-3"

Ex: -

```
<div class="row border m-0">
   <div class="col border text-bg-primary order-5 order-md-3">col1</div>
   <div class="col border text-bg-success order-1 order-md-5">col2</div>
   <div class="col border text-bg-danger order-3 order-md-1">col3</div>
```

```
</div>
```

Offset: - Move columns to the right using .offset-breakpoint-number classes. These classes increase the left margin of a column by number of columns.

Syntax: -

Ex: -

```
<div class="row m-0">
   <div class="col-2 border">col1</div>
   <div class="col-4 offset-md-2 border">col2</div>
   <div class="col-2 offset-md-2 border">col3</div>
</div>
```

Grid gap: - it set gap between row and columns.

Gap: - it set gap between all items in grid with .gap class, with breakpoints and 0 to 5 sizes.

Syntax: - gap-breakpoint-size.

Ex: -

```
<div class="row gap-sm-5">
   <div class="p-2 col-5 border">Grid item 1</div>
   <div class="p-2 col-5 border">Grid item 2</div>
   <div class="p-2 col-5 border">Grid item 3</div>
   <div class="p-2 col-5 border">Grid item 4</div>
</div>
```

Row gap: - it set row gap between all items in grid with .row-gap class, with breakpoints and 0 to 5 sizes.

Syntax: - row-gap-breakpoint-size.

Ex: -

```
<div class="row row-gap-sm-5">
   <div class="p-2 col-6 border">Grid item 1</div>
   <div class="p-2 col-6 border">Grid item 2</div>
   <div class="p-2 col-6 border">Grid item 3</div>
   <div class="p-2 col-6 border">Grid item 4</div>
</div>
```

Column gap: - it set column gap between all items in grid with .column-gap class, with breakpoints and 0 to 5 sizes.

Syntax: - column-gap-breakpoint-size.

Ex: -

```
<div class="row column-gap-sm-5">
   <div class="p-2 col-5 border">Grid item 1</div>
   <div class="p-2 col-5 border">Grid item 2</div>
```

```
  <div class="p-2 col-5 border">Grid item 3</div>
  <div class="p-2 col-5 border">Grid item 4</div>
</div>
```

Bootstrap CSS: -

Image: -

Image fluid: - it set max width: 100% and height: auto for scale with parent
Ex: - class='img-fluid'
<img src='../3css/res/background.jpg' class='img-fluid'>

Image thumbnails: - it gives 1px rounded border to image
Ex: - class='img-thumbnail'
<img src='../3css/res/background.jpg' class='img-thumbnail'>

Table: -

Table: - set table style
Ex: - class='table'
<table class='table'>

Table color: - set color of table, it can also particularly set to table row or cell.
Ex: - class='table-success'
Type: - primary, secondary, success, danger, warning, info, light, dark
<table class='table table-success'>

Striped row: - it adds zebra-striping to any table row within the tbody
Ex: - class='table-striped'
<table class='table table-striped table-dark'>

Striped column: - it adds zebra-striping to any table column
Ex: - class='table-striped-columns'
<table class='table table-striped-columns table-dark'>

Table hover: - it enables a hover state on table row within the tbody
Ex: - class='table-hover'
<table class='table table-hover table-dark'>

Table active: - it highlights a table row or cell
Ex: - class='table-active'
<tr class='table-active'>
<td class='table-active'>

Table bordered: - it set border to table, use border-color property
Ex: - class='table-bordered'
```
<table class='table table-bordered border-dark'>
```

Table borderless: - it set table without borders
Ex: - class='table-borderless'
```
<table class='table table-borderless'>
```

Small table: - it make table more compact by cutting all cell padding in half
Ex: - class='table-sm'
```
<table class='table table-sm'>
```

Table group divider: - add thicker and darker border between table groups tbody, thead and tfoot.
Ex: - class='table-group-divider'
```
<tbody class='table-group-divider'>
```

Caption top: - add caption to top of table
Ex: - class='caption-top'
```
<table class='table caption-top'>
```

Table responsive: - it allows table scroll horizontally with ease, also can add breakpoint, it will set responsive up to breakpoint
Ex: - class='table-responsive-md'
```
<div class='table-responsive-md'>
   <table class='table' style='width: 900px'>
   … … …
   </table>
</div>
```

Background: -

Transparent: - it set background transparent
Ex: - class='bg-transparent'
```
<div class='bg-primary p-5'>
   <div class='bg-transparent border'>GOOD DAY</div>
</div>
```

Gradient: - it set gradient background also need to apply bg-color or text-bg-color classes
Ex: - class='text-bg-success bg-gradient'
```
<div class='text-bg-success bg-gradient p-5'> good day </div>
```

Opacity: - it set opacity of background color

Class: - bg-opacity-75, bg-opacity-50, bg-opacity-25, bg-opacity-10

Ex: - class='bg-success p-5 bg-opacity-10'

```
<div class='bg-success bg-opacity-10 p-5'>
   good day
</div>
```

Font: -

Font size: - set font size with font size class.

Classes: - fs-1 to fs-6

Ex: -

```
<span class='fs-1'>good</span>
```

Font weight: - set font weight

Classes: - fw-bold, fw-bolder, fw-semibold, fw-medium, fw-normal, fw-light, fw-lighter

Ex: -

```
<span class='fw-bold fs-1'>good</span>
```

Font italic: - set font style to italic or normal

Classes: - fst-italic, fst-normal

Ex: -

```
<span class='fst-italic fs-1'>good</span>
```

Line height: - set line height

Classes: - lh-1, lh-sm, lh-base, lh-lg

Ex: -

```
<p class='lh-lg'> On the Insert … … … specify directly</p>
```

Font monospace: - set monospace font

Ex: -

```
<p class='font-monospace'> On the Insert … … … specify directly</p>
```

Text: -

Text reset: - it reset link’s color for inherit text color from parent element

Ex: -

```
<p class="text-success">
  Text is good here <a href="#" class='text-reset'>click link</a>.
</p>
```

Text decoration: - it decorate text

Classes: - text-decoration-underline, text-decoration-line-through, text-decoration-none

Ex: -

```
<p class='text-decoration-underline'>Good day to all</p>
```

```
<p class='text-decoration-line-through'>Good day to all</p>
<p><a href='#' class='text-decoration-none'>Good day to all</a></p>
```

Opacity: - it control opacity of element
Classes: - opacity-100, opacity-75, opacity-50, opacity-25, opacity-0
Ex: -

```
<div class='p-5 border text-bg-primary w-25 opacity-50'>good day</div>
```

Link: - use classes with link

Link opacity: - it set transparency of link and opacity when hover link
Classes: - link-opacity-100, link-opacity-75, link-opacity-50, link-opacity-25, link-opacity-10, link-opacity-10-hover, link-opacity-25-hover, link-opacity-50-hover, link-opacity-75-hover, link-opacity-100-hover
Ex: -

```
<p><a href='#' class='link-opacity-75 link-opacity-25-hover'>GOOD day</a></p>
```

Link underline color: - it set underline color of link
Classes: - link-underline-primary, link-underline-secondary, link-underline-success, link-underline-danger, link-underline-warning, link-underline-info, link-underline-light, link-underline-dark
Ex: -

```
<p><a href='#' class='link-underline-success'>GOOD day</a></p>
```

Link underline offset: - it set underline distance from link text
Classes: - link-offset-1, link-offset-2, link-offset-3
Ex: -

```
<p><a href='#' class='link-offset-3'>GOOD day</a></p>
```

Link underline opacity: - it set opacity of link underline
Classes: - link-underline-opacity-0, link-underline-opacity-10, link-underline-opacity-25, link-underline-opacity-50, link-underline-opacity-75, link-underline-opacity-100
Ex: - use ‘link-underline’ class for work ‘link-underline-opacity-*’ class properly

```
<p><a href='#' class='link-underline link-underline-opacity-0'>GOOD day</a></p>
```

Link hover variants: - change link style using hover variants.
Classes: - link-opacity-*-hover, link-offset-*-hover, link-underline-opacity-*-hover
Ex: -

```
<p><a href='#' class='link-underline link-offset-3-hover link-underline-opacity-50-hover link-opacity-75-hover'>GOOD day</a></p>
```

Colored link: - it change link of colors
Classes: - link-primary, link-secondary, link-success, link-danger, link-warning, link-info, link-light, link-dark

Ex: -

```
<p><a href='#' class='link-success'>GOOD day</a></p>
```

Interaction: - it set that how user will interact with website content

Text selection: - it set way of content selection

Classes: - user-select-all, user-select-none, user-select-auto

Ex: -

```
<p class="user-select-all">this paragraph will be entirely selected when clicked by the user. </p>
<p class="user-select-auto">this paragraph has default select behavior. </p>
<p class="user-select-none">this paragraph will not be selectable when clicked by the user. </p>
```

Pointer event: - it provides class for add or prevent interaction with element.

Classes: - pe-none, pe-auto

Note: - note assign tabindex='-1' attribute for avoid keyboard interaction, assign aria-disabled='true' for declare element is not operable or editable in case of assistive technologies as screen reader

Ex: -

```
<p><a href='#' class='pe-none' tabindex='-1' aria-disabled='true'>Good day</a></p>
<p  class='pe-none'>
   <a href='#' tabindex='-1' aria-disabled='true'>Good day (Disable link) </a>
   <a href='#' class='pe-auto'>Good day (Enable link) </a>
</p>
```

Media setup: -

Object fit: - it resizes of img and video element’s content to fit it in container, also can add breakpoints

Classes: - object-fit-contain, object-fit-cover, object-fit-fill, object-fit-scale, object-fit-none, object-fit-sm-contain, object-fit-md-cover, etc

Ex: -

```
<img src='../3css/res/background.jpg' height='300px' width='400px' class='border rounded object-fit-xl-none object-fit-sm-contain'>
```

Ratio: - it set aspect ratio for handling images and videos

Classes: - ratio, ratio-16x9, ratio-1x1, ratio-4x3, ratio-21x9

Ex: -

```
<div class='border ratio ratio-21x9' style='width:300px;'>
   <div class='bg-primary'></div>
</div>
```

Custom ratios: - also can set custom ratio with CSS property

Property: - --bs-aspect-ratio: 50%; //use percentage

Ex: -

```
<div class='border ratio' style='width:300px; --bs-aspect-ratio: 50%;'>
```

```
   <div class='bg-primary'></div>
</div>
```

Custom breakpoint: - also can set custom breakpoint with CSS

Ex: -

CSS style: -

```
.ratio-4x3{
   @include media-breakpoint-up(md) {
   --bs-aspect-ratio: 50%; // 2x1
   }
}
```

Html: -

```
<div class='border ratio ratio-4x3' style='width: 300px; '>
   <div class='text-bg-primary'>Good</div>
</div>
```

Vertical align: - it change alignment of elements, vertical-align only effects inline, inline-block, inline-table and table cell elements.

Classes: - align-baseline, align-middle, align-top, align-bottom, align-text-bottom, align-text-top

Ex: -

```
<table style="height: 100px;" border="1px">
   <tbody>
      <tr>
         <td class="align-baseline border">baseline</td>
         <td class="align-middle border">middle</td>
         <td class="align-top border">top</td>
         <td class="align-bottom border">bottom</td>
      </tr>
   </tbody>
</table>

<p class='border'>
   <span class='align-baseline'>go(baseline)</span>
   <span class='align-middle'>go(middle)</span>
   H<span class='align-text-bottom' style='font-size: 0.7em'>2</span>O
   10<span class='align-text-top' style='font-size: 0.7em'>th</span>
</p>
```

Element design: -

Shadow: - add or remove shadows to elements with box-shadow utilities

Classes: - shadow-none, shadow-sm, shadow, shadow-lg

Ex: -

```
<div class='container p-3 border bg-body-tertiary shadow'>GOOD DAY</div>
```

Overflow: - it set content to overflows element. Also can set horizontal and vertical overflow

Classes: - overflow-auto, overflow-hidden, overflow-visible, overflow-scroll, overflow-x-auto, overflow-x-*, overflow-y-auto, overflow-y-*

Ex: -

```
<div class='border overflow-scroll p-2' style='width: 400px; height: 400px;'>
   <p style='width: 600px;'>On the …
   … … …
   … current template.</p>
</div>
```

Text truncation: - it truncate long string of text

Class: - 'text-truncate

Ex: -

```
<div style='width: 600px;' class='text-truncate'> to change … … … current template. </div>
```

Position: - set position of elements

Classes: - position-static, position-relative, position-absolute, position-fixed, position-sticky

Arrange elements: - these classes use 0, 50 and 100 values for percentage

Classes: - top-0, end-50, start-100, bottom-50

Ex: -

```
<div class='p-3 text-bg-primary text-center position-sticky bottom-50 start-50' style='width: 120px;'
>GOOD DAY</div>

<div class='col-12 border position-relative' style='height: 300px;'>
   <div class='text-bg-primary p-1 position-absolute top-0 start-50'>good</div>
</div>
```

Translate middle: - this class apply translateX(-50%) and translateY(-50%) to element, also can translate on X or Y axis only.

Classes: - translate-middle, translate-middle-x, translate-middle-y

Ex: -

```
<div class='border position-relative text-center' style='width: 500px; height: 300px;'>GOOD
   <div class='border position-absolute top-50 start-50 translate-middle' style='width:50px; height:
50px;'> DAY</div>
</div>
```

Fixed helper: - use classes for quickly configure element to top and bottom position

Classes: - fixed-top, fixed-bottom

Ex: -

```
<div class='container-fluid text-bg-primary fixed-top' style='height: 90px;'>MENU</div>
<div class='container-fluid text-bg-primary fixed-bottom' style='height: 50px;'>FOOTER</div>
```

Sticky helper: - it quickly configure sticky element, also set responsive configuration

Classes: - sticky-top, sticky-bottom, sticky-md-top, sticky-xl-bottom, etc

Ex: -

```
<div class='container-fluid text-bg-primary sticky-top' style='height: 50px;'>Menu</div>
<div class='container-fluid text-bg-primary sticky-xl-bottom' style='height: 50px;'>FOOTER</div>
```

Z-index: - it set stack level of element or component

Classes: - z-n1, z-0, z-1, z-2, z-3

Ex: -

```
<div class='container-fluid p-0 fixed-top z-n1' style='height: 100%;'>
   <img class='img-fluid' src='../3css/res/background.jpg' width="100%">
</div>
```

Bootstrap z-index: - bootstrap set some default z-index for some component overlays

zindex-dropdown	1000	zindex-modal-backdrop	1050
zindex-sticky	1020	zindex-modal	1055
zindex-fixed	1030	zindex-popover	1070
zindex-offcanvas-backdrop	1040	zindex-tooltip	1080
zindex-offcanvas	1045	zindex-toast	1090

Display: - it set display property for quickly toggle display value responsively, also can set breakpoint classes

Classes: - d-none, d-inline, d-inline-block, d-block, d-sm-inline, d-md-block, d-xl-none, etc.

Ex: -

```
<div class="text-bg-primary p-2 d-none d-sm-inline d-md-none">GOOD</div>
```

Display print: - display or hide elements in print preview mode

Classes: - d-print-none, d-print-inline, d-print-inline-block, d-print-block

Ex: -

```
<div class="text-bg-danger text-center p-2 d-none d-print-block">
   Design is Copyright reserved
</div>
```

Visually hidden: - it hides elements visually and focusable with screen reader or keyboard tab

Classes: - visually-hidden, visually-hidden-focusable

Ex: -

```
<h2 class="visually-hidden">Title for screen readers</h2>
<a class="visually-hidden-focusable" href="#content">Skip to main content</a>
<div class="visually-hidden-focusable">
     A container with a <a href="#">focusable element</a>.
</div>
```

Visibility: - it set visibility of elements.
Classes: - visible, invisible
Ex: -
<div class='container bg-success border invisible' style='height: 200px;'></div>

Focus ring: -
it set custom focus ring around focusable elements, with classes or custom properties
Classes: - focus-ring, focus-ring-primary, focus-ring-success, focus-ring-dark, etc.
Ex: -
<a href='#go' class='text-bg-success focus-ring focus-ring-danger' id='go'>click here</a>

Custom properties: - it set custom styling to focus ring in style.
Properties: - --bs-focus-ring-color: red, --bs-focus-ring-width: 10px, --bs-focus-ring-opacity: 0.50, --bs-focus-ring-x: 10px, --bs-focus-ring-y: 10px, --bs-focus-ring-blur: 4px.
Ex: -

```
<a href='#go' class='focus-ring' style='--bs-focus-ring-color: red;' id='go' >click here</a>
<a href='#go' class='focus-ring' style='--bs-focus-ring-width: 10px; --bs-focus-ring-opacity: 0.95;' id='go'>click here</a>
<a href='#go' class='focus-ring text-bg-danger'style='--bs-focus-ring-x: 10px; --bs-focus-ring-y: 10px; --bs-focus-ring-blur: 5px;' id='go'>click here</a>
```

Stretched link: -
it set html or bootstrap element clickable by nested link, for stretch, set position relative of container element
Classes: - stretched-link
Ex: -

```
<div class='text-bg-success p-2 position-relative' style='width: 12rem; '>
  <p class='p-2 border'>Here is link <a href='#' class='link-light stretched-link'>GO to</a></p>
</div>
```

Ex: - for stretched link in particular area, set position relative to nearest container element

```
<div class='text-bg-success p-2' style='width: 12rem; '>
  <img src='../3css/res/background.jpg' class='img-fluid'>
  <p class='p-2 border position-relative'>
    Here is link <a href='#' class='link-light stretched-link'> Butterfly</a>
  </p>
</div>
```

Flex: -
set flex property with display class, also can set break-point with flex property.
Classes: - d-flex, d-inline-flex, d-md-flex, d-sm-inline-flex

Direction: - set direction of flex item, and also can set break-point with it.
Classes: - flex-row, flex-row-reverse, flex-column, flex-column-reverse, flex-md-row, flex-sm-column, etc

Ex: -

```
<div class='text-bg-success border d-flex flex-column flex-md-row'>
   <div class='p-2 border'>Good day1</div>
   <div class='p-2 border'>Good day2</div>
   <div class='p-2 border'>Good day3</div>
</div>
```

Justify content: - it change alignment of flex-items on main axis (x-axis) if flex direction row or on cross axis (y-axis) if flex direction is column, also can set breakpoint

Classes: - justify-content-start, justify-content-end, justify-content-center, justify-content-between, justify-content-around, justify-content-evenly, justify-content-sm-start, justify-content-md-center, justify-content-lg-evenly, etc.

Ex: -

```
<div class='text-bg-success border d-flex justify-content-start justify-content-md-evenly'>
   <div class='p-2 border'>Good day1</div>
   <div class='p-2 border'>Good day2</div>
   <div class='p-2 border'>Good day3</div>
</div>
```

Align items: - it change alignment of flex-items on cross axis (y-axis) if flex direction row or on main axis (x-axis) if flex direction is column, also can set breakpoint

Classes: - align-items-start, align-items-center, align-items-end, align-items-baseline, align-items-stretch, align-items-sm-baseline, align-items-md-stretch, etc.

Ex: -

```
<div class='text-bg-success border d-flex align-items-stretch align-items-sm-baseline' style='height: 150px;'>
   <div class='p-2 border fs-3'>Good day1</div>
   <div class='p-2 border'>Good day2</div>
   <div class='p-2 border'>Good day3</div>
</div>
```

Align self: - it set alignment of individual flex-item in flex-box on cross-axis (y-axis) if flex-direction is row and on main axis (x-axis) if flex-direction is column. Also can set breakpoints

Classes: - align-self-start, align-self-end, align-self-center, align-self-baseline, align-self-stretch, align-self-sm-start, align-self-lg-end, etc.

Ex: -

```
<div class='text-bg-success border d-flex' style='height: 150px;'>
   <div class='p-2 border'>Good day1</div>
   <div class='p-2 border align-self-center'>Good day2</div>
   <div class='p-2 border'>Good day3</div>
</div>
```

Fill: - it force flex-item to take equal widths to their content or equal width if content does not surpass their border box, while taking up all available horizontal space. Also can set breakpoints

Classes: - flex-fill, flex-sm-fill, flex-md-fill, etc.

Ex: -

```
<div class='text-bg-success border d-flex'>
   <div class='p-2 border flex-fill'>Very Good day</div>
   <div class='p-2 border flex-fill'>Good day2</div>
   <div class='p-2 border flex-fill'>Good day3</div>
</div>
```

Grow: - it fills available space by grow flex item. It grows with flex-grow-1, also can set breakpoints

Classes: - flex-grow-1, flex-grow-0, flex-sm-grow-1, flex-sm-grow-0, flex-md-grow-1, etc.

Ex: -

```
<div class='text-bg-success border d-flex'>
   <div class='p-2 border flex-grow-1'>Very Good day</div>
   <div class='p-2 border flex-sm-grow-1'>Good day2</div>
   <div class='p-2 border'>Good day3</div>
</div>
```

Shrink: - it shrink flex item if necessary. It shrink with flex-hrink-1, also can set breakpoints

Classes: - flex-shrink-1, flex-shrink-0, flex-sm-shrink-1, flex-sm-shrink-0, flex-md-shrink-1, etc.

Ex: -

```
<div class='text-bg-success border d-flex'>
   <div class='p-2 border w-100'>Very Good day</div>
   <div class='p-2 border flex-shrink-1'>Good day2</div>
</div>
```

Auto margin in flex-box: - it pushes items in left or right when margin auto class is applies to flex item.

Classes: - mx-auto, ms-auto, me-auto

Ex: -

```
<div class='text-bg-success border d-flex'>
   <div class='p-2 border me-auto'>Good day1</div>
   <div class='p-2 border'>Good day2</div>
   <div class='p-2 border'>Good day3</div>
</div>
```

Auto margin with align-items: - it move flex-items vertically with align-items and flex-column using auto margin-top and bottom

Classes: - my-auto, mt-auto, mb-auto

Ex: -

```
<div class='text-bg-success border d-flex align-items-start flex-column' style='height: 200px;'>
   <div class='p-2 border'>Good day1</div>
   <div class='p-2 border mt-auto'>Good day2</div>
   <div class='p-2 border '>Good day3</div>
</div>
```

Wrap: - it set flex-items wrap in flex container, also can set breakpoint

Classes: - flex-wrap, flex-nowrap, flex-wrap-reverse, flex-sm-wrap, flex-lg-wrap-reverse, etc.

Ex: -

```
<div class='text-bg-success border d-flex flex-wrap flex-xl-nowrap'>
   <div class='p-2 border'>Good day1</div>
   … … …
   <div class='p-2 border'>Good day20</div>
</div>
```

Order: - it set appearance order of flex-items in flex-box from 0 to 5 also order first and order last is also applicable. Also can set breakpoint

Classes: - order-first, order-0, order-1, order-2, order-3, order-4, order-5, order-last, order-sm-1, order-lg-last, etc.

Ex: -

```
<div class='text-bg-success border d-flex'>
   <div class='p-2 border'>Good day1</div>
   <div class='p-2 border order-4 order-md-last'>Good day2</div>
   <div class='p-2 border order-2 order-md-first'>Good day3</div>
</div>
```

Align content: - it aligns flex items together on the cross axis (y-axis). It show effect when flex-items are wrap. Also can set breakpoint

Classes: - align-content-start, align-content-center, align-content-end, align-content-between, align-content-around, align-content-stretch, align-content-md-stretch, align-content-sm-around, etc.

Ex: -

```
<div class='text-bg-success border d-flex align-content-stretch flex-wrap' style='height: 200px;'>
   <div class='p-2 border'>Good day1</div>
   <div class='p-2 border'>Good day20</div>
</div>
```

Flex uses: -

Media object: - create media object with flex classes.

Ex: -

```
<div class="d-flex align-items-center p-3 border text-bg-success">
   <div class="flex-shrink-0">
      <img src="../3css/res/background.jpg" class="rounded" style="height: 200px;">
   </div>
   <div class="flex-grow-1 ms-3">
      This is background images of butterflies on it.
   </div>
</div>
```

Stack: - it helps to build flex-box utilities to make component layout faster.

Vertical: - create vertical layouts and use gap class for create gap between them.

Class: - vstack

Ex: -

```
<div class='text-bg-success py-2 border vstack gap-3'>
   <div class='p-2 border text-bg-primary'>Good day1</div>
   <div class='p-2 border text-bg-primary'>Good day2</div>
   <div class='p-2 border text-bg-primary'>Good day3</div>
</div>
```

Horizontal: - create horizontal layouts and use gap class for create gap between them.

Class: - hstack

Ex: -

```
<div class='text-bg-success p-2 border hstack gap-3'>
   <div class='p-2 border text-bg-primary'>Good day1</div>
   <div class='p-2 border text-bg-primary'>Good day2</div>
   <div class='p-2 border text-bg-primary'>Good day3</div>
</div>
```

Vertical rule: - it is vertical divider like <hr> element.

Class: - vr

Ex: -

```
<div class='vr'></div>
```

Ex: - use vertical rule in horizontal stack

```
<div class='text-bg-success px-2 border hstack gap-3'>
   <div class='p-2 border text-bg-primary'>Good day1</div>
   <div class='p-2 border text-bg-primary ms-auto'>Good day2</div>
   <div class='vr opacity-100'></div>
   <div class='p-2 border text-bg-primary'>Good day3</div>
</div>
```

Icons: - bootstrap have own icon library for try.

Install: -

1. Go to bootstrap icon page and click on install.
2. Copy CDN link for html or CSS.
3. Click on icon then copy icon font and paste it in html.
4. Also can change icons size, color, background colors with font properties or utilities.

Ex: -

```
<!DOCTYPE html>
```

```
<html>
   <head>
      <title> ICONS </title>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
      <link href='https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css'
rel='stylesheet' integrity='sha384-
QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH'
crossorigin='anonymous'>
      <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstrap-
icons.min.css">
      <style>
         *{
            margin: 0px;
            padding: 0px;
         }
      </style>
   </head>
   <body>

      <i class="bi bi-alarm text-bg-dark"></i>
      <i class="bi bi-xbox fs-3 text-success"></i>

      <script src='https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js'
integrity='sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz'
crossorigin='anonymous'></script>
      <script src='https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.8/dist/umd/popper.min.js'
integrity='sha384-I7E8VVD/ismYTF4hNIPjVp/Zjvgyol6VFvRkX/vR+Vc4jQkC+hVqc2pM8ODewa9r'
crossorigin='anonymous'></script>
   </body>
</html>
```

Icon link: - it create stylized hyperlinks with icons

Class: - icon-link

Ex: -

```
<a class="icon-link" href="#">
   <i class="bi bi-box-seam"></i> Icon link
</a>
```

Icon link hover: - move icon to the right on hover

```
<a class="icon-link icon-link-hover" href="#">
   Icon link<i class="bi bi-arrow-right"></i>
</a>
```

Change transformation: - customize transform on hover by overriding transform variable for link.

Variable: - --bs-icon-link-transform
Ex: -

```
<a class="icon-link icon-link-hover" style="--bs-icon-link-transform: translate(0, -0.3em)" href="#">
  <i class="bi bi-clipboard"></i> Icon link
</a>
```

Change color on hover: - change color on hover with link hover color variable.
Variable: - --bs-link-hover-color-rgb
Ex: -

```
<a class="icon-link icon-link-hover" style="--bs-link-hover-color-rgb: 55, 135, 84;" href="#">
  <i class="bi bi-clipboard"></i>Icon link
</a>
```

Bootstrap Components (1): - there are pre-build components in bootstrap for use

Buttons: - it set style to element like button. It made for <button> element, but also can use with <input> or <a> elements, set role="button" for <a> element for appropriately work with assistive technology as screen readers.

Classes: - btn
Ex: -

```
<button type='button' class='btn'>Good</button>
```

Buttons Colors: - also can set colors to buttons
Classes: - btn-primary, btn-secondary, btn-success, btn-danger, btn-warning, btn-info, btn-light, btn-dark
Ex: -

```
<button type='button' class='btn btn-primary'>Good</button>
<input type='button' class='btn btn-success' value='Good'>
<a href='#' role='button' class='btn btn-dark'>Good</a>
```

Outline buttons: - style button with outline.
Classes: - btn-outline-primary, btn-outline-secondary, btn-outline-success, btn-outline-danger, btn-outline-warning, btn-outline-info, btn-outline-light, btn-outline-dark
Ex: -

```
<a href='#' role='button' class='btn btn-outline-dark'>Good</a>
```

Buttons size: - change the size of buttons
Classes: - btn-sm, btn-lg
Ex: -

```
<button type='button' class='btn btn-primary btn-sm'>Good</button>
```

Custom button: - customize button style with css variables

Variables: - --bs-btn-padding-y, --bs-btn-padding-x, --bs-btn-font-size

Ex: -

```
<button type='button' class='btn btn-primary' style="--bs-btn-padding-y: 0.1em; --bs-btn-padding-x:
1.2em; --bs-btn-font-size: 1.2em;">Custom</button>
```

Buttons disabled state: - use disabled attribute with buttons elements

Ex: -

```
<button type='button' class='btn btn-primary disabled'>Good</button>
```

Disabled link button: - disable link with disabled class, set tab-index -1 and area-disabled as true

Ex: -

```
<a href='#' role='button' class='btn btn-primary disabled' tabindex="-1" aria-disabled="true">Good</a>
```

Block buttons: - create block button of responsive full width

Ex: -

```
<div class="d-grid gap-2">
   <button type='button' class='btn btn-primary'>Good</button>
   <button type='button' class='btn btn-primary'>Good</button>
</div>
```

Ex: - change block buttons in simple buttons

```
<div class="d-grid gap-2 d-md-block">
   <button type='button' class='btn btn-primary'>Good</button>
   <button type='button' class='btn btn-primary'>Good</button>
</div>
```

Ex: - adjust block button width and center block button

```
<div class="d-grid gap-2 col-6 mx-auto">
   <button type='button' class='btn btn-primary'>Good</button>
   <button type='button' class='btn btn-primary'>Good</button>
</div>
```

Ex: - display block buttons on small screen and shift them right on big screen

```
<div class="d-grid gap-2 d-md-flex justify-content-md-end">
   <button type='button' class='btn btn-primary'>Good</button>
   <button type='button' class='btn btn-primary'>Good</button>
</div>
```

Toggle states: - add data-bs-toggle='button' attribute for toggle to active state, also can add .active class for show pre-active button and add aria-pressed='true' for assistive technology.

Ex: -

```
<button type='button' class='btn btn-outline-dark' data-bs-toggle="button">Good</button>
```

```
<button type='button' class='btn btn-outline-dark active' data-bs-toggle="button" aria-pressed='true'>Good</button>
```

Toggle link button: -

Ex: -

```
<a href='#' role='button' class='btn' data-bs-toggle='button'>Good</a>
<a href='#' role='button' class='btn active' data-bs-toggle='button' aria-pressed='true'>Good</a>
```

Button group: - group series of buttons together. Set role='group' and aria-label for naming button group, for assistive technology

Class: - btn-group

Ex: -

```
<div class="btn-group" role='group' aria-label='group of buttons'>
   <button type="button" class="btn btn-primary">b1</button>
   <button type="button" class="btn btn-primary">b2</button>
   <button type="button" class="btn btn-primary">b3</button>
</div>
```

Mixed style: -

Ex: -

```
<div class="btn-group" role='group' aria-label='group of buttons'>
   <button type="button" class="btn btn-danger">b1</button>
   <button type="button" class="btn btn-warning">b2</button>
   <button type="button" class="btn btn-success">b3</button>
</div>
```

Outline style: -

Ex: -

```
<div class="btn-group" role='group' aria-label='group of buttons'>
   <button type="button" class="btn btn-outline-info">b1</button>
   <button type="button" class="btn btn-outline-info">b2</button>
   <button type="button" class="btn btn-outline-info">b3</button>
</div>
```

Checkbox button group: - pair checkbox input to label with class .btn, and set input class .btn-check for hide input element

Pair: - use for attribute in label and passed id of input to for attribute, it will focus on input when label is clicked

Ex: -

```
<div class="btn-group" role="group">
   <input type="checkbox" class="btn-check" id="ch1">
   <label class="btn btn-outline-primary" for="ch1">Checkbox 1</label>
```

```
  <input type="checkbox" class="btn-check" id="ch2">
  <label class="btn btn-outline-primary" for="ch2">Checkbox 2</label>
</div>
```

Radio button group: - use pairing and .btn-check class as checkbox

Ex: -

```
<div class="btn-group" role="group">
  <input type="radio" class="btn-check" name="rdo" id="r1" checked>
  <label class="btn btn-outline-primary" for="r1">Radio 1</label>

  <input type="radio" class="btn-check" name="rdo" id="r2">
  <label class="btn btn-outline-primary" for="r2">Radio 2</label>
</div>
```

Button toolbar: - combine sets of button groups into button toolbar, set role="toolbar" for assistive technology

Classes: - btn-toolbar

Ex: -

```
<div class="btn-toolbar" role="toolbar" aria-label="buttons toolbar">
  <div class="btn-group me-2" role="group" aria-label="First group">
    <button type="button" class="btn btn-primary">1</button>
    <button type="button" class="btn btn-primary">2</button>
    <button type="button" class="btn btn-primary">3</button>
  </div>
  <div class="btn-group me-2" role="group" aria-label="Second group">
    <button type="button" class="btn btn-secondary">4</button>
    <button type="button" class="btn btn-secondary">5</button>
  </div>
</div>
```

Button group sizing: - set size of button group

Classes: - btn-group-lg, btn-group-sm

Ex: -

```
<div class="btn-group btn-group-sm" role="group">
  <button type="button" class="btn btn-primary">1</button>
  <button type="button" class="btn btn-primary">2</button>
</div>
```

Nesting button group: - place .btn-group within anthor .btn-group

Ex: -

```
<div class="btn-group" role="group">
  <button type="button" class="btn btn-primary">p1</button>
  <button type="button" class="btn btn-primary">p2</button>
  <div class="btn-group btn-group-sm" role="group">
    <button type="button" class="btn btn-info">i1</button>
    <button type="button" class="btn btn-info">i2</button>
  </div>
</div>
```

Vertical button group: - set button group vertically

Classes: - btn-group-vertical

Ex: -

```
<div class="btn-group-vertical" role="group">
  <button type="button" class="btn btn-primary">1</button>
  <button type="button" class="btn btn-primary">2</button>
  <button type="button" class="btn btn-primary">3</button>
</div>
```

Collapse: - hide content of page and display when click. Use data-bs-toggle='collapse' is required, also set data-bs-target with button element, set id of element as value that applied .collapse class to target.

Also can use href with link button

Classes: - collapse

Ex: -

```
<a type='button' class='btn btn-primary' data-bs-toggle='collapse' role='button' href='#box'>
  Click here (link)
</a>
<button type='button' class='btn btn-primary' data-bs-toggle='collapse' data-bs-target='#box'>
  Click here (button)
</button>
<div class='collapse border m-2 p-2' id='box'>
    Good day
</div>
```

Horizontal collapse: - it collapse and show element horizontal way.

Classes: - collapse-horizontal

Ex: -

```
<button type='button' class='btn btn-primary' data-bs-toggle='collapse' data-bs-target='#box2'>
  Click here (button)
</button>
<div style='width: 300px;'>
  <div class='collapse collapse-horizontal m-2 p-2 border' id='box2'>
    Good day
  </div>
</div>
```

Multiple toggles and target: - use class selector for target multiple element

Ex: -

```
<a type='button' class='btn btn-primary' data-bs-toggle='collapse' role='button' href='.boxes'>
   Click here (link)
</a>
<button type='button' class='btn btn-primary' data-bs-toggle='collapse' data-bs-target='.boxes'>
   Click here (button)
</button>
<div class='row'>
   <div class='collapse border col-6 p-2 boxes'>
      Good day 1
   </div>
   <div class='collapse border col-6 p-2 boxes'>
      Good day 2
   </div>
</div>
```

Navs and tabs: - navigation components of bootstrap

Base nav: - it share general markup and style. Use class .nav for define container as navigation, use .nav-item to define container of links as navtionation items and .nav-link for styling links in .nav or .nav-item container. Use .active for set current page and .disabled for disabled link, also can use aria-current='page' for active link-page and aria-disabled='true' for disabled link, for assistive technology.

Classes: - nav, nav-item, nav-link, active, disabled

Ex: - navigation with ul, li and link element

```
<ul class='nav'>
   <li class='nav-item'>
      <a href='#' class='nav-link active' aria-current='page'>Home</a>
   </li>
   <li class='nav-item'>
      <a href='#' class='nav-link'>About</a>
   </li>
   <li class='nav-item'>
      <a href='#' class='nav-link disabled' aria-disabled='true'>Private</a>
   </li>
</ul>
```

Ex: - navigation with nav tag and links

```
<nav class='nav'>
   <a href='#' class='nav-link active' aria-current='page'>Home</a>
   <a href='#' class='nav-link'>Help</a>
   <a href='#' class='nav-link disabled' aria-disabled='true'>About</a>
</nav>
```

Horizontal alignment of nav: - change the horizontal alignment of nav, also can set breakpoint

Classes: - justify-content-center, justify-content-end, justify-content-md-end

Ex: -

```
<nav class='nav justify-content-center'>
   <a href='#' class='nav-link active' aria-current='page'>Home</a>
   <a href='#' class='nav-link'>Help</a>
   <a href='#' class='nav-link disabled' aria-disabled='true'>About</a>
</nav>
```

Set vertical nav: - change nav to vertical with .flex-column also can set breakpoint.

Classes: - flex-column, flex-md-column

Ex: -

```
<nav class='nav flex-md-column'>
   <a href='#' class='nav-link active' aria-current='page'>Home</a>
   <a href='#' class='nav-link'>Help</a>
   <a href='#' class='nav-link disabled' aria-disabled='true'>About</a>
</nav>
```

Tabs: - it generate tabbed interface, show tabs as active

Class: - nav-tabs

Ex: -

```
<nav class='nav nav-tabs'>
   <a href='#' class='nav-link' aria-current='page'>Home</a>
   <a href='#' class='nav-link active'>Help</a>
   <a href='#' class='nav-link disabled' aria-disabled='true'>about</a>
</nav>
```

Pills: - generate pill on active link

Class: - nav-pills

Ex: -

```
<nav class='nav nav-pills'>
   <a href='#' class='nav-link active' aria-current='page'>Home</a>
   <a href='#' class='nav-link'>Help</a>
   <a href='#' class='nav-link disabled' aria-disabled='true'>about</a>
</nav>
```

Underlines: - generate underline on active link

Class: - nav-underline

Ex: -

```
<nav class='nav nav-underline'>
   <a href='#' class='nav-link active' aria-current='page'>Home</a>
   <a href='#' class='nav-link'>Help</a>
   <a href='#' class='nav-link disabled' aria-disabled='true'>About</a>
```

```
</nav>
```

Fill and justify: - extend nav in full available width, nav items takes space according to content, also can justify nav items with equal width from available.

Class: - nav-fill, nav-justified

Ex: -

```
<nav class='nav nav-underline nav-fill' style='width: 500px;'>
  <a href='#' class='nav-link active' aria-current='page'>Home</a>
  <a href='#' class='nav-link'>Help</a>
  <a href='#' class='nav-link disabled' aria-disabled='true'>About</a>
</nav>
```

Change content when click on nav: - for it add bootstrap JavaScript’s bundle and popper link html file

Create .nav and button or link with data-bs-toggle="tab" and data-bs-target with target id in .tab-content’s element with .tab-pane class. Set .show and .active for first element container in .tab-pane, also can set .fade for fade effect.

Classes: - tab-content, tab-pane, fade, show, active

Ex: -

```
<nav class="nav nav-pills">
  <a href="#" class="nav-link active" data-bs-toggle="tab" data-bs-target="#homec">Home</a>
  <a href="#" class="nav-link" data-bs-toggle="tab" data-bs-target="#aboutc">About</a>
  <a href="#" class="nav-link disabled" data-bs-toggle="tab" data-bs-target="#disabledc"> Disabled
</a>
</nav>

<div class="tab-content">
  <div class="tab-pane fade show active" id="homec" tabindex='0'>
    Home Content
  </div>
  <div class="tab-pane fade" id="aboutc" tabindex='0'>
    About Content
  </div>
  <div class="tab-pane fade" id="disabledc" tabindex='0'>
    Disabled Content
  </div>
</div>
```

Vertical nav with changeable content: - set nav and changing content with vertical nav bar, take container element with .d-flex and .align-items-start class and set nav with .flex-column and element with .tab-content in that container element

Ex: -

```
<div class='d-flex align-items-start'>
```

```
  <nav class="nav nav-pills flex-column">
    <a href="#" class="nav-link active" data-bs-toggle="tab" data-bs-target="#homevc">Home</a>
    <a href="#" class="nav-link" data-bs-toggle="tab" data-bs-target="#aboutvc">About</a>
    <a href="#" class="nav-link disabled" data-bs-toggle="tab" data-bs-target="#disvc">Disabled</a>
  </nav>

  <div class="tab-content">
    <div class="tab-pane fade show active" id="homevc" tabindex='0'>
      Home Content
    </div>
    <div class="tab-pane fade" id="aboutvc" tabindex='0'>
      About Content
    </div>
    <div class="tab-pane fade" id="disvc" tabindex='0'>
      Disabled Content
    </div>
  </div>
</div>
```

Dropdowns: - it toggle contextual overlays for displaying lists of links and more. Set button with data-bs-toggle='dropdown' attribute, use container element with .dropdown-menu and elements inside it with .dropdown-item, and wrap button and .dropdown-menu container in element with .dropdown class. Also can use .dropdown-toggle for button to show dropdown arrow on button and set button colors

Classes: - dropdown, dropdown-toggle, dropdown-menu, dropdown-item

Ex: -

```
<div class='dropdown'>
  <button class='btn btn-success dropdown-toggle' data-bs-toggle='dropdown'>Dropdown</button>
  <ul class='dropdown-menu'>
    <li><a class='dropdown-item' href='#'>Good day1</a></li>
    <li><a class='dropdown-item' href='#'>Good day2</a></li>
    <li><a class='dropdown-item' href='#'>Good day3</a></li>
  </ul>
</div>
```

Split button: - set button split with dropdown, wrap two buttons and container with .dropdown-menu in element with .btn-group, first button is with text and second button with .dropdown-toggle and .dropdown-toggle-split for proper spacing around dropdown.

Class: - dropdown-toggle-split

Ex: -

```
<div class='btn-group'>
  <button class='btn btn-success'>Dropdown</button>
```

```
   <button class='btn btn-success dropdown-toggle dropdown-toggle-split' data-bs-toggle='dropdown'>
</button>
   <ul class='dropdown-menu'>
      <li><a class='dropdown-item' href='#'>Good day1</a></li>
      <li><a class='dropdown-item' href='#'>Good day2</a></li>
      <li><div class='dropdown-divider'></div></li>
      <li><a class='dropdown-item' href='#'>Good day3</a></li>
   </ul>
</div>
```

Dropdown sizing: - also can set size of dropdown buttons with size classes for buttons.

Classes: - btn-lg, btn-sm

Ex: -

```
<div class='btn-group'>
   <button class='btn btn-success btn-lg'>Dropdown</button>
   <button class='btn btn-success btn-lg dropdown-toggle dropdown-toggle-split' data-bs-toggle=
'dropdown'></button>
   <ul class='dropdown-menu'>
      <li><a class='dropdown-item' href='#'>Good day1</a></li>
      <li><a class='dropdown-item' href='#'>Good day2</a></li>
      <li><div class='dropdown-divider'></div></li>
      <li><a class='dropdown-item' href='#'>Good day3</a></li>
   </ul>
</div>
```

Dropdown centered: - make dropdown menu centered below toggle button.

Class: - 'dropdown-center

Ex: -

```
<div class='dropdown-center'>
   <button class='btn btn-success dropdown-toggle' data-bs-toggle='dropdown'>Dropdown</button>
   <ul class='dropdown-menu'>
      <li><a class='dropdown-item' href='#'>Good day1</a></li>
      <li><a class='dropdown-item' href='#'>Good day2</a></li>
      <li><a class='dropdown-item' href='#'>Good day3</a></li>
   </ul>
</div>
```

Drop-up: - show dropdown menus above elements. Also can center drop-up menu on toggle element

Classes: - dropup, dropup-center

Ex: -

```
<div class='btn-group dropup dropup-center'>
   <button class='btn btn-success'>Dropdown</button>
```

```
  <button class='btn btn-success dropdown-toggle dropdown-toggle-split' data-bs-toggle='dropdown'></button>
  <ul class='dropdown-menu'>
    <li><a class='dropdown-item' href='#'>Good day1</a></li>
    <li><a class='dropdown-item' href='#'>Good day2</a></li>
    <li><a class='dropdown-item' href='#'>Good day3</a></li>
  </ul>
</div>
```

Drop-end: - show dropdown menus at the right of the element.

Class: - dropend

Ex: -

```
<div class='btn-group dropend'>
  <button class='btn btn-success'>Dropdown</button>
  <button class='btn btn-success dropdown-toggle dropdown-toggle-split' data-bs-toggle='dropdown'></button>
  <ul class='dropdown-menu'>
    <li><a class='dropdown-item' href='#'>Good day1</a></li>
    <li><a class='dropdown-item' href='#'>Good day2</a></li>
    <li><a class='dropdown-item' href='#'>Good day3</a></li>
  </ul>
</div>
```

Drop-start: - show dropdown menus at the left of the element. Add text button after container of .dropdown-menu element.

Classes: - dropstart

Ex: -

```
<div class='btn-group dropstart'>
  <button class='btn btn-success dropdown-toggle dropdown-toggle-split' data-bs-toggle='dropdown'></button>
  <ul class='dropdown-menu'>
    <li><a class='dropdown-item' href='#'>Good day1</a></li>
    <li><a class='dropdown-item' href='#'>Good day2</a></li>
    <li><a class='dropdown-item' href='#'>Good day3</a></li>
  </ul>
  <button class='btn btn-success'>Dropdown</button>
</div>
```

Menu items: - use <a> and <button> elements for dropdown items, also can set non-interactive items

Class: - dropdown-item-text

Ex: -

```
<div class="dropdown">
```

```
   <button class="btn btn-secondary dropdown-toggle" type="button" data-bs-toggle="dropdown"
aria-expanded="false">Dropdown</button>
   <ul class="dropdown-menu">
      <li><button class="dropdown-item" type="button">Action</button></li>
      <li><span class="dropdown-item-text">Action 2</span></li>
      <li><button class="dropdown-item" type="button">Another action</button></li>
      <li><button class="dropdown-item" type="button">Something else here</button></li>
   </ul>
</div>
```

Active and disabled: - set dropdown item's style as active or disabled

Classes: - active, disabled

Ex: -

```
<div class="dropdown">
   <button class="btn btn-secondary dropdown-toggle" type="button" data-bs-toggle="dropdown"
aria-expanded="false">Dropdown</button>
   <ul class="dropdown-menu">
      <li><a class="dropdown-item" href="#">Regular link</a></li>
      <li><a class="dropdown-item active" href="#" aria-current="true">Active link</a></li>
      <li><a class="dropdown-item disabled" href="#" aria-current="true">Disabled link</a></li>
      <li><a class="dropdown-item" href="#">Another link</a></li>
   </ul>
</div>
```

Menu alignment: - set dropdown menu align to left or right side, also can set breakpoints, set classes with .dropdown-menu

Classes: - dropdown-menu-end, dropdown-menu-start, dropdown-menu-md-end, dropdown-menu-xl-start

Ex: -

```
<div class="dropdown">
   <button class="btn btn-secondary dropdown-toggle container-fluid" type="button" data-bs-
toggle="dropdown" aria-expanded="false"> Dropdown </button>
   <ul class="dropdown-menu dropdown-menu-md-end dropdown-menu-xl-start">
      <li><a class="dropdown-item" href="#">Regular link</a></li>
      <li><a class="dropdown-item" href="#">Active link</a></li>
      <li><a class="dropdown-item" href="#">Another link</a></li>
   </ul>
</div>
```

Menu headers and dividers: - set dropdown header and dividers in .dropdown-menu

Classes: - dropdown-header, dropdown-divider

Ex: -

```
<div class="dropdown">
```

```
  <button class="btn btn-secondary dropdown-toggle" data-bs-toggle="dropdown">Dropdown
</button>
  <ul class="dropdown-menu">
    <li><h4 class="dropdown-header">Header text</h4></li>
    <li><a class="dropdown-item" href="#">Regular link</a></li>
    <li><div class="dropdown-divider"></div></li>
    <li><a class="dropdown-item" href="#">Another link</a></li>
  </ul>
</div>
```

Dropdown offset: - use data-bs-offset to change location of dropdown-menu
Attribute: - data-bs-offset='x, y', data-bs-offset='60, 5'
Ex: -

```
<div class="dropdown">
  <button class="btn btn-secondary dropdown-toggle" data-bs-toggle="dropdown" data-bs-
offset='60,5'>Dropdown</button>
  <ul class="dropdown-menu">
    <li><a class="dropdown-item" href="#">Regular link</a></li>
    <li><a class="dropdown-item" href="#">Regular2 link</a></li>
    <li><a class="dropdown-item" href="#">Another link</a></li>
  </ul>
</div>
```

Dropdown reference: - use data-bs-reference to change position of dropdown-menu display
Attribute: - data-bs-reference="child" //child, parent
Ex: -

```
<div class="btn-group">
  <button class="btn btn-secondary">Dropdown</button>
  <button class="btn btn-secondary dropdown-toggle" data-bs-toggle="dropdown" data-bs-reference
="child" data-bs-offset='30,10'></button>
  <ul class="dropdown-menu">
    <li><a class="dropdown-item" href="#">Regular link</a></li>
    <li><a class="dropdown-item" href="#">Regular2 link</a></li>
    <li><a class="dropdown-item" href="#">Another link</a></li>
  </ul>
</div>
```

Dropdown auto-close: - change behavior of .dropdown-menu when to close.
Attribute: - data-bs-auto-close='true' // true, inside, outside, false
Ex: -

```
<div class="dropdown">
  <button class="btn btn-secondary dropdown-toggle" data-bs-toggle="dropdown" data-bs-auto-close
='inside'>Dropdown</button>
```

```
    <ul class="dropdown-menu">
        <li><a class="dropdown-item" href="#">Regular link</a></li>
        <li><a class="dropdown-item" href="#">Regular2 link</a></li>
        <li><a class="dropdown-item" href="#">Another link</a></li>
    </ul>
</div>
```

Dropdown with nav: - show dropdown in nav

Ex: -

```
<ul class='nav nav-tabs'>
    <li class='nav-item'>
        <a class='nav-link active' href='#'>Home</a>
    </li>
    <li class='nav-item dropdown'>
        <a class='nav-link dropdown-toggle' data-bs-toggle='dropdown' href='#'>Courses</a>
        <ul class='dropdown-menu'>
            <li><a class='dropdown-item' href='#'>C++</a></li>
            <li><a class='dropdown-item' href='#'>JAVA</a></li>
            <li><a class='dropdown-item' href='#'>PYTHON</a></li>
        </ul>
    </li>
    <li class='nav-item'>
        <a class='nav-link' href='#'>About</a>
    </li>
    <li class='nav-item'>
        <a class='nav-link disabled' href='#'>Help</a>
    </li>
</ul>
```

Bootstrap Forms: - use for styling form in bootstrap

Form basic: -

Classes: - form-label, form-control

Ex: -

```
<div class="p-3 border m-3 rounded">
    <label class="form-label" for="emailinp">Email</label>
    <input type="email" class="form-control" id="emailinp" placeholder="enter email">
</div>
```

Sizing: - set height of inputs

Classes: - form-control-lg, form-control-sm

Ex: -

```
<input class="form-control form-control-lg" type="text" placeholder=".form-control-lg">
```

```
<input class="form-control" type="text" placeholder="Default input">
<input class="form-control form-control-sm" type="text" placeholder=".form-control-sm">
```

Form text: - show text associated to form input
Class: - form-text
Ex: -

```
<div class="row g-3 align-item-center">
   <div class="col-auto">
      <label for="pass" class="col-form-label">Password</label>
   </div>
   <div class="col-auto">
      <input type="password" class="form-control" id="pass">
   </div>
   <div class="col-auto">
      <div class="form-text"> write password here </div>
   </div>
</div>
```

Disabled and readonly: - these are Boolean attributes for disabled input or set it readonly

Attributes: - disabled, readonly
Ex: -

```
<input type="text" class="form-control" value="GOOD DAY" disabled>
<input type="text" class="form-control" value="GOOD DAY" readonly>
```

Readonly plain text: - display readonly input as plain text
Class: - form-control-plaintext
Ex: -

```
<input type="text" class="form-control-plaintext" value="GOOD DAY" readonly>
```

Input: -

File input: - set style to file input
Ex: -

```
<label for="sfile" class="form-label">Select single image</label>
<input type="file" id="sfile" class="form-control">
<label for="mfile" class="form-label">Select multiple files</label>
<input type="file" id="mfile" class="form-control" multiple>
```

Color input: - set styling of color input
Classes: - form-control-color
Ex: -

```
<label for="colorpick" class="form-label">Color picker</label>
<input type="color" id="colorpick" class="form-control form-control-color" value="#ff0000">
```

Datalist: - set datalist style

Ex: -

```
<label for="dtl">Enter choice</label>
<input list="choice" id="dtl" class="form-control" placeholder="Type here">
<datalist id="choice">
   <option value="choice1">
   <option value="choice2">
   <option value="choice3">
</datalist>
```

Select: - change style with bootstrap

Class: - form-select

Ex: -

```
<select class="form-select">
   <option selected>Select any</option>
   <option value="1">one</option>
   <option value="2">two</option>
   <option value="3">three</option>
</select>
```

Sizing: - set size of select

Classes: - form-select-lg, form-select-sm

Ex: -

```
<select class="form-select form-select-lg">
   <option selected>Select any</option>
   <option value="1">one</option>
   <option value="2">two</option>
   <option value="3">three</option>
</select>
```

Multiple select: - set multiple attribute for select multiple options, also can set size attribute with value for show options

Attribute: - multiple, size="2" / positive integer value

Ex: -

```
<select class="form-select form-select-sm" multiple size="2">
   <option selected>Select any</option>
   <option value="1">one</option>
   <option value="2">two</option>
   <option value="3">three</option>
</select>
```

Disabled select: - set disabled attribute to select

Attribute: - disabled

Ex: -

```
<select class="form-select" disabled>
   <option selected>Select any</option>
   <option value="1">one</option>
   <option value="2">two</option>
   <option value="3">three</option>
</select>
```

Checkbox and radio: - set style of checkbox and radio buttons

Classes: - form-check, form-check-input, form-check-label

Ex: -

```
<div class="form-check">
   <input class="form-check-input" type="checkbox" id="ch1">
   <label class="form-check-label" for="ch1">Checkbox1</label>
</div>
```

Disabled checkbox: - set style for disabled checkbox

Ex: -

```
<div class="form-check">
   <input class="form-check-input" type="checkbox" id="ch3" disabled checked>
   <label class="form-check-label" for="ch3">Checkbox3</label>
</div>
```

Radio: -

Ex: -

```
<div class="form-check">
   <input class="form-check-input" type="radio" id="r1" name="rg">
   <label class="form-check-label" for="r1">radio1</label><br>
</div>
```

Disabled radio: -

Ex: -

```
<div class="form-check">
   <input class="form-check-input" type="radio" id="r2" name="rg" disabled>
   <label class="form-check-label" for="r2">radio2</label>
</div>
```

Switches: - style checkbox as switch

Class: - form-switch

Ex: -

```
<div class="form-check form-switch">
   <input class="form-check-input" type="checkbox" id="sw1">
   <label class="form-check-label" for="sw1">switch1</label>
</div>
```

Disabled switch: -

Ex: -

```
<div class="form-check form-switch">
   <input class="form-check-input" type="checkbox" id="sw2" disabled>
   <label class="form-check-label" for="sw2">switch2</label>
</div>
```

Inline inputs: - and group of inputs horizontally

Class: - form-check-inline

Ex: -

```
<div class="form-check form-check-inline">
   <input class="form-check-input" type="checkbox" id="c1">
   <label class="form-check-label" for="c1">check1</label>
</div>
<div class="form-check form-check-inline">
   <input class="form-check-input" type="checkbox" id="c2">
   <label class="form-check-label" for="c2">check2</label>
</div>
```

Reverse input: - set input right side of container

Class: - form-check-reverse

Ex: -

```
<div class="form-check form-check-reverse">
   <input class="form-check-input" type="radio" id="rb1" name="rgr">
   <label class="form-check-label" for="rb1">radio1</label>
</div>
<div class="form-check form-check-reverse">
   <input class="form-check-input" type="radio" id="rb2" name="rgr">
   <label class="form-check-label" for="rb2">radio2</label>
</div>
```

Toggle buttons: - create toggle button with checkbox or radio and .btn on label, set .btn-check to input for hide input and for link it with label.

Classes: - btn-check, btn, btn-primary

Ex: - checkbox

```
<input type="checkbox" class="btn-check" id="t1">
<label class="btn btn-primary" for="t1">Toggle1</label>
```

```
<input type="checkbox" class="btn-check" id="t2" checked>
<label class="btn btn-primary" for="t2">Toggle2</label>

<input type="checkbox" class="btn-check" id="t3" disabled>
<label class="btn btn-primary" for="t3">Toggle3</label>
```

Ex: - radio button

```
<input type="radio" class="btn-check" id="rt1" name="choice">
<label class="btn" for="rt1">Toggle1</label>

<input type="radio" class="btn-check" id="rt2" name="choice" checked>
<label class="btn" for="rt2">Toggle2</label>

<input type="radio" class="btn-check" id="rt3" name="choice" disabled>
<label class="btn" for="rt3">Toggle3</label>

<input type="radio" class="btn-check" id="rt4" name="choice">
<label class="btn" for="rt4">Toggle4</label>
```

Range: - style range input with bootstrap

Class: - form-range

Ex: -

```
<label for="range1" class="form-label">Example range</label>
<input type="range" class="form-range" id="range1">
```

Disabled: - disabled range input

Ex: -

```
<label for="range2" class="form-label">range2</label>
<input type="range" class="form-range" id="range2" disabled>
```

Min and max: - define min and max values default is 0 and 100 respectively

Ex: -

```
<label for="range3" class="form-label">range3</label>
<input type="range" class="form-range" id="range3" min="0" max="10">
```

Steps: - change step value

Ex: -

```
<label for="range4" class="form-label">range4</label>
<input type="range" class="form-range" id="range4" min="0" max="3" step="0.5">
```

Input group: - it set group of text and buttons with inputs

Classes: - input-group, input-group-text

Ex: -

```
<div class="input-group">
   <label class="input-group-text" for="ig1">@</label>
   <input type="text" class="form-control" id="ig1">
</div>
```

Sizing input group: - set size of input groups

Classes: - input-group-sm, input-group-lg

Ex: -

```
<div class="input-group input-group-sm">
   <label class="input-group-text" for="ig1">@</label>
   <input type="text" class="form-control" id="ig1">
</div>
```

Checkbox and radio button in input group: - use .form-check-input and .mt-0 when there is no visible text next to input

Ex: -

```
<div class="input-group">
   <div class="input-group-text">
      <input type="checkbox" class="form-check-input mt-0">
   </div>
   <input type="text" class="form-control">
</div>
```

Multiple inputs: - set multiple input in one group

Ex: -

```
<div class="input-group">
   <span class="input-group-text">Full name</span>
   <input type="text" class="form-control" placeholder="first name">
   <input type="text" class="form-control" placeholder="last name">
</div>
```

Multiple add-ons: -

Ex: -

```
<div class="input-group">
   <span class="input-group-text">$</span>
   <span class="input-group-text">@</span>
   <input type="text" class="form-control">
</div>
```

Button add-ons: -

Ex: -

```
<div class="input-group">
```

```
   <input type="search" class="form-control" placeholder="Search here">
   <button type="button" class="btn btn-primary">Search</button>
</div>
```

Buttons with dropdown: -

Ex: -

```
<div class="input-group">
   <input type="text" class="form-control">
   <button class="btn btn-outline-secondary dropdown-toggle" data-bs-toggle="dropdown">Dropdown
</button>
   <ul class="dropdown-menu dropdown-menu-end">
      <li><a class="dropdown-item" href="#">Action 1</a></li>
      <li><a class="dropdown-item" href="#">Action 2</a></li>
      <li><hr class="dropdown-divider"></li>
      <li><a class="dropdown-item" href="#">Separate Action</a></li>
   </ul>
</div>
```

Segmented buttons: -

Ex: -

```
<div class="input-group">
   <input type="text" class="form-control">
   <button class="btn btn-outline-secondary">Action</button>
   <button class="btn btn-outline-secondary dropdown-toggle dropdown-toggle-split" data-bs-toggle=
"dropdown"></button>
   <ul class="dropdown-menu dropdown-menu-end">
      <li><a class="dropdown-item" href="#">Action 1</a></li>
      <li><a class="dropdown-item" href="#">Action 2</a></li>
      <li><hr class="dropdown-divider"></li>
      <li><a class="dropdown-item" href="#">Separate Action</a></li>
   </ul>
</div>
```

Custom forms: - input groups include supports for custom selects and custom file inputs

Custom select: -

Ex: -

```
<div class="input-group">
   <select class="form-select">
      <option selected>Choose...</option>
      <option value="1">One</option>
      <option value="2">Two</option>
      <option value="3">Three</option>
   </select>
```

```
    <button class="btn btn-outline-secondary">Button</button>
</div>
```

Custom file input: -

Ex: -

```
<div class="input-group">
    <input type="file" class="form-control">
    <button class="btn btn-outline-secondary" type="button">Button</button>
</div>
```

Floating labels: -
labels that float over your input fields, set place holder for work properly

Class: - form-floating

Ex: -

```
<div class="form-floating">
    <input type="email" class="form-control" placeholder="enter email">
    <label>Email address</label>
</div>
```

Text-area floating label: -

Ex: -

```
<div class="form-floating">
    <textarea class="form-control" placeholder="write comments" style="height: 100px;"></textarea>
    <label>Comments</label>
</div>
```

Select floating label: -

Ex: -

```
<div class="form-floating">
    <select class="form-select">
        <option selected>select</option>
        <option value="1">One</option>
        <option value="2">Two</option>
        <option value="3">Three</option>
    </select>
    <label>Select any option</label>
</div>
```

Disabled floating label: -

Ex: -

```
<div class="form-floating">
    <input type="text" class="form-control" placeholder="disable" disabled>
    <label>Disabled</label>
</div>
```

Read-only plaintext floating label: -

Ex: -

```
<div class="form-floating">
   <input type="email" readonly class="form-control-plaintext" placeholder="email">
   <label>Empty input</label>
</div>
```

Input group floating label: -

Ex: -

```
<div class="input-group">
   <span class="input-group-text">$</span>
   <div class="form-floating">
      <input type="text" class="form-control" placeholder="enter username">
      <label>Username</label>
   </div>
</div>
```

Form layout: - set structure of form

Margin utilities: - set form structure with margins

Ex: -

```
<div class="mb-3 mx-3">
   <label for="lay1" class="form-label">Username</label>
   <input type="text" class="form-control" id="lay1" placeholder="Enter username">
</div>
<div class="mb-3 mx-3">
   <label for="lay2" class="form-label">Password</label>
   <input type="pass" class="form-control" id="lay2" placeholder="Enter password">
</div>
```

Form grid: - set form using grid classes

Ex: -

```
<div class="row">
   <div class="col">
      <input type="text" class="form-control" placeholder="First name">
   </div>
   <div class="col">
      <input type="text" class="form-control" placeholder="Last name">
   </div>
</div>
```

Form gutter classes: - also can set gutter classes

Ex: -

```
<div class="row g-3">
   <div class="col">
      <input type="text" class="form-control" placeholder="First name">
   </div>
   <div class="col">
      <input type="text" class="form-control" placeholder="Last name">
   </div>
</div>
```

Horizontal form: -

Ex: -

```
<div class="row">
   <label class="col-sm-2 col-form-label" for="em1">Email</label>
   <div class="col-sm-10">
      <input type="email" class="form-control" id="em1">
   </div>
</div>
<div class="row">
   <label class="col-sm-2 col-form-label" for="ps1">password</label>
   <div class="col-sm-10">
      <input type="password" class="form-control" id="ps1">
   </div>
</div>
```

Horizontal form label sizing: - set label size according to input

Classes: - col-form-label-lg, col-form-label-sm

Ex: -

```
<div class="row mb-3">
   <label class="col-sm-2 col-form-label col-form-label-sm" for="em1">Email</label>
   <div class="col-sm-10">
      <input type="email" class="form-control form-control-sm" id="em1">
   </div>
</div>
<div class="row">
   <label class="col-sm-2 col-form-label col-form-label-lg" for="ps1">password</label>
   <div class="col-sm-10">
      <input type="password" class="form-control form-control-lg" id="ps1">
   </div>
</div>
```

Column sizing in layout: - set column size with .col utilities

Ex: -

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

```
  <div class="col-sm-7">
    <input type="text" class="form-control" placeholder="City">
  </div>
  <div class="col-sm">
    <input type="text" class="form-control" placeholder="State">
  </div>
  <div class="col-sm">
    <input type="text" class="form-control" placeholder="Zip">
  </div>
</div>
```

Auto column sizing: -

Ex: -

```
<form class="row align-items-center">
  <div class="col-auto">
    <input type="text" class="form-control" placeholder="Name">
  </div>
  <div class="col-auto">
    <div class="input-group">
      <div class="input-group-text">@</div>
      <input type="text" class="form-control" placeholder="Username">
    </div>
  </div>
  <div class="col-auto">
    <div class="form-check">
      <input class="form-check-input" type="checkbox" id="rm">
      <label class="form-check-label" for="rm">Remember me</label>
    </div>
  </div>
  <div class="col-auto">
    <button type="submit" class="btn btn-primary">Submit</button>
  </div>
</form>
```

Inline forms: - use .row-cols-* classes to create responsive horizontal

Classes: - row-cols-lg-auto

Ex: -

```
<form class="row row-cols-lg-auto align-items-center">
  <div class="col-12">
    <div class="input-group">
      <div class="input-group-text">@</div>
      <input type="text" class="form-control" placeholder="Username">
    </div>
  </div>
  <div class="col-12">
    <div class="form-check">
      <input class="form-check-input" type="checkbox" id="rem">
      <label class="form-check-label" for=" rem ">Remember me</label>
    </div>
  </div>
  <div class="col-12">
    <button type="submit" class="btn btn-primary">Submit</button>
  </div>
</form>
```

Bootstrap Components (2): - components in bootstrap

Breadcrumb: - indicates current page location within a navigational hierarchy that automatically adds separators via CSS

Classes: - breadcrumb, breadcrumb-item, active

Ex: -

```
<nav>
  <ol class="breadcrumb">
    <li class="breadcrumb-item active"><a href="#">Home</a></li>
    <li class="breadcrumb-item"><a href="#">library</a></li>
    <li class="breadcrumb-item">help</li>
  </ol>
</nav>
```

Divider: - change dividers in breadcrumb

Variable: - --bs-breadcrumb-divider: '|' //values '%', '*', '@', '!', '>', none etc

Ex: -

```
<nav style="--bs-breadcrumb-divider: '|';">
  <ol class="breadcrumb">
    <li class="breadcrumb-item active"><a href="#">Home</a></li>
    <li class="breadcrumb-item"><a href="#">library</a></li>
    <li class="breadcrumb-item">help</li>
  </ol>
</nav>
```

Badges: - it is small count and labeling component

Class: - badge

Heading: - set badge with heading

Ex: -

```
<h1>Example heading <span class="badge text-bg-secondary">New</span></h1>
```

Buttons: - set badge on button

Ex: -

```
<button type="button" class="btn btn-primary">
  Notifications <span class="badge text-bg-secondary">4</span>
</button>
```

Positioned: - set position of badge

Class: - rounded-pill //set rounded pill badge

Ex: -

```
<button type="button" class="btn btn-primary position-relative">
  Inbox
  <span class="position-absolute top-0 start-100 translate-middle badge rounded-pill bg-danger">
    99+
  </span>
</button>
```

Badge with rounded circle: -

Class: - rounded-circle //set badge circle

Ex: -

```
<button type="button" class="btn mx-3 btn-primary position-relative">
  Profile
  <span class="position-absolute top-0 start-100 translate-middle p-2 bg-danger border border-light
rounded-circle"></span>
</button>
```

Background color: - set background colors to badge

Ex: -

```
<span class="badge text-bg-primary">Primary</span>
```

Pill badges: - set badges as rounded pill

Ex: -

```
<span class="badge rounded-pill text-bg-primary">Primary</span>
```

List group: - it is flexible and powerful component for displaying series of content

Classes: - list-group, list-group-item

Ex: -

```
<ul class="list-group">
  <li class="list-group-item">An item</li>
  <li class="list-group-item">A second item</li>
  <li class="list-group-item">A third item</li>
</ul>
```

Active and disabled: - add .active to show active list group item and .disabled to show disable list group item

Class: - active, disabled

Ex: -

```
<ul class="list-group">
   <li class="list-group-item active">An item</li>
   <li class="list-group-item">A second item</li>
   <li class="list-group-item disabled">A third item</li>
</ul>
```

Links and buttons: - create actionable list group items

Ex: -

```
<div class="list-group">
   <a href="#" class="list-group-item list-group-item-action active">An item</a>
   <a href="#" class="list-group-item list-group-item-action">A second item</a>
   <button type="button" class="list-group-item list-group-item-action">A third item</button>
   <button type="button" class="list-group-item list-group-item-action">A fourth item</button>
</div>
```

Flush: - remove rounded corners and borders of list group items

Class: - list-group-flush

Ex: -

```
<ul class="list-group list-group-flush">
   <li class="list-group-item">An item</li>
   <li class="list-group-item">A second item</li>
   <li class="list-group-item">A third item</li>
</ul>
```

Numbered: - set numbers to list-group-items

Class: - list-group-numbered

Ex: -

```
<ul class="list-group list-group-numbered">
   <li class="list-group-item">An item</li>
   <li class="list-group-item">A second item</li>
   <li class="list-group-item">A third item</li>
</ul>
```

List group with custom content: -

Ex: -

```
<ul class="list-group list-group-numbered col-3">
   <li class="list-group-item d-flex justify-content-between align-items-start">
      <div class="ms-2 me-auto">
```

```
        <div class="fw-bold">First</div>
       First list item
     </div>
     <span class="badge text-bg-primary rounded-pill">14</span>
  </li>
  <li class="list-group-item d-flex justify-content-between align-items-start">
     <div class="ms-2 me-auto">
        <div class="fw-bold">Second</div>
           Second list item
        </div>
        <span class="badge text-bg-primary rounded-pill">17</span>
  </li>
  <li class="list-group-item d-flex justify-content-between align-items-start">
     <div class="ms-2 me-auto">
        <div class="fw-bold">Third</div>
           Third list item
        </div>
        <span class="badge text-bg-primary rounded-pill">10</span>
  </li>
</ul>
```

Horizontal list group: - set list group items in horizontal position also can add break points. Add .flex-fill to each list group item for equal width when horizontal

Class: - list-group-horizontal, list-group-horizontal-md, etc

Ex: -

```
<ul class="list-group list-group-horizontal-md">
  <li class="list-group-item flex-fill">An item</li>
  <li class="list-group-item flex-fill">A second item</li>
  <li class="list-group-item flex-fill">A third item</li>
</ul>
```

List group item variants: - set colors of list group items

Classes: - list-group-item-primary, list-group-item-seconday, list-group-item-danger, etc

Ex: -

```
<ul class="list-group">
  <li class="list-group-item">A simple default list group item</li>
  <li class="list-group-item list-group-item-primary">A simple primary list group item</li>
  <li class="list-group-item list-group-item-secondary">A simple secondary list group item</li>
  <li class="list-group-item list-group-item-success">A simple success list group item</li>
  <li class="list-group-item list-group-item-danger">A simple danger list group item</li>
  <li class="list-group-item list-group-item-warning">A simple warning list group item</li>
  <li class="list-group-item list-group-item-info">A simple info list group item</li>
  <li class="list-group-item list-group-item-light">A simple light list group item</li>
  <li class="list-group-item list-group-item-dark">A simple dark list group item</li>
</ul>
```

List group with badges: - add badges to list groups

Ex: -

```
<ul class="list-group col-4">
  <li class="list-group-item d-flex justify-content-between align-items-center">
    A list item
    <span class="badge text-bg-primary rounded-pill">14</span>
  </li>
  <li class="list-group-item d-flex justify-content-between align-items-center">
    A second list item
    <span class="badge text-bg-primary rounded-pill">2</span>
  </li>
  <li class="list-group-item d-flex justify-content-between align-items-center">
    A third list item
    <span class="badge text-bg-primary rounded-pill">1</span>
  </li>
</ul>
```

Custom content: -

Ex: -

```
<div class="list-group">
 <a href="#" class="list-group-item list-group-item-action active">
  <div class="d-flex w-100 justify-content-between">
   <h5 class="mb-1">List group item heading</h5>
   <small>3 days ago</small>
  </div>
  <p class="mb-1">Some placeholder content in a paragraph.</p>
  <small>And some small print.</small>
 </a>
 <a href="#" class="list-group-item list-group-item-action">
  <div class="d-flex w-100 justify-content-between">
   <h5 class="mb-1">List group item heading</h5>
   <small class="text-body-secondary">3 days ago</small>
  </div>
  <p class="mb-1">Some placeholder content in a paragraph.</p>
  <small class="text-body-secondary">And some muted small print.</small>
 </a>
 <a href="#" class="list-group-item list-group-item-action">
  <div class="d-flex w-100 justify-content-between">
   <h5 class="mb-1">List group item heading</h5>
   <small class="text-body-secondary">3 days ago</small>
```

```
    </div>
    <p class="mb-1">Some placeholder content in a paragraph.</p>
    <small class="text-body-secondary">And some muted small print.</small>
  </a>
</div>
```

Checkboxes in list group: -

Ex: -

```
<ul class="list-group">
    <li class="list-group-item">
        <input class="form-check-input me-1" type="checkbox" id="lch1">
        <label class="form-check-label" for="lch1">First checkbox</label>
    </li>
    <li class="list-group-item">
        <input class="form-check-input me-1" type="checkbox" id="lch2">
        <label class="form-check-label" for="lch2">Second checkbox</label>
    </li>
    <li class="list-group-item">
        <input class="form-check-input me-1" type="checkbox" id="lch3">
        <label class="form-check-label" for="lch3">Third checkbox</label>
    </li>
</ul>
```

Radio buttons in list group: -

Ex: -

```
<ul class="list-group">
    <li class="list-group-item">
        <input class="form-check-input me-1" type="radio" name="lgr" id="lgr1" checked>
        <label class="form-check-label" for="lgr1">First radio</label>
    </li>
    <li class="list-group-item">
        <input class="form-check-input me-1" type="radio" name="lgr" id="lgr2">
        <label class="form-check-label" for="lgr2">Second radio</label>
    </li>
    <li class="list-group-item">
        <input class="form-check-input me-1" type="radio" name="lgr" id="lgr3">
        <label class="form-check-label" for="lgr3">Third radio</label>
    </li>
</ul>
```

Stretched link in link group items: - set .stretched-link to label for make link group item clickable

Ex: -

```
<ul class="list-group">
```

```
  <li class="list-group-item list-group-item-action">
    <input class="form-check-input me-1" type="checkbox" id="cb1">
    <label class="form-check-label stretched-link" for="cb1">First checkbox</label>
  </li>
  <li class="list-group-item list-group-item-action">
    <input class="form-check-input me-1" type="checkbox" id="cb2">
    <label class="form-check-label stretched-link" for="cb2">Second checkbox</label>
  </li>
  <li class="list-group-item list-group-item-action">
    <input class="form-check-input me-1" type="checkbox" id="cb3">
    <label class="form-check-label stretched-link" for="cb3">Third checkbox</label>
  </li>
</ul>
```

Javascript behavior: - use javascript plug-in for create panes of local content that change on click. set data-bs-toggle value as list in .list-group-item, also set anchor element as .list-group-item and target id of .tab-pane container for show content on click. Also can set .fade to container for fade effect

Attribute: - data-bs-toggle="list"

Classes: - tab-content, tab-pane, fade, show active

Ex: -

```
<div class="row">
  <div class="col-4">
    <div class="list-group">
      <a class="list-group-item list-group-item-action active" data-bs-toggle="list" href="#list-home">
        Home
      </a>
      <a class="list-group-item list-group-item-action" data-bs-toggle="list" href="#list-profile">
        Profile
      </a>
      <a class="list-group-item list-group-item-action" data-bs-toggle="list" href="#list-messages">
        Messages
      </a>
    </div>
  </div>
  <div class="col-8">
    <div class="tab-content">
      <div class="tab-pane fade show active" id="list-home">
        Home content
      </div>
      <div class="tab-pane fade" id="list-profile">
        Profile content
      </div>
      <div class="tab-pane fade" id="list-messages">
```

```
        Message content
      </div>
    </div>
  </div>
</div>
```

Cards: - it provide a flexible and extensible content container with multiple variants

Classes: - card, card-img-top, card-body, card-title, card-text

Ex: -

```
<div class="card" style="width: 18rem;">
  <img src="../3css/res/background.jpg" class="card-img-top">
  <div class="card-body">
    <h5 class="card-title">Card title</h5>
    <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>
    <a href="#" class="btn btn-primary">Go somewhere</a>
  </div>
</div>
```

Card content types: -

Body: - set body of card, wrap .card-body container in .card conainer

Classes: - card, card-body

Ex: -

```
<div class="card" style="width: 18rem;">
  <div class="card-body">
    This is some text within a card body.
  </div>
</div>
```

Titles, text and links: - set card title, text and links in .card-body container

Classes: - card-title, card-subtitle, card-text, card-link

Ex: -

```
<div class="card m-2" style="width: 18rem;">
  <div class="card-body">
    <h5 class="card-title">Card title</h5>
    <h6 class="card-subtitle mb-2 text-body-secondary">Card subtitle</h6>
    <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>
    <a href="#" class="card-link">Card link</a>
    <a href="#" class="card-link">Another link</a>
  </div>
```

```
</div>
```

Images in card: - set images in card, it set top and bottom corners rounded to match the card's border

Classes: - card-img-top, card-img-bottom

Ex: -

```
<div class="card m-2" style="width: 18rem;">
   <img src="../3css/res/background.jpg" class="card-img-top">
   <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>
```

List group in card: - set list group in card

Classes: - card-header, card-footer

Ex: -

```
<div class="card" style="width: 18rem;">
   <div class="card-header">
      Card header
   </div>
   <ul class="list-group list-group-flush">
      <li class="list-group-item">An item</li>
      <li class="list-group-item">A second item</li>
      <li class="list-group-item">A third item</li>
   </ul>
   <div class="card-footer">
      Card footer
   </div>
</div>
```

Kitchen sink: - Mix and match multiple content types to create the card you need, or throw everything in there.

Ex: -

```
<div class="card" style="width: 18rem;">
   <img src="..." class="card-img-top" alt="...">
   <div class="card-body">
      <h5 class="card-title">Card title</h5>
      <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>
```

```
  <ul class="list-group list-group-flush">
    <li class="list-group-item">An item</li>
    <li class="list-group-item">A second item</li>
    <li class="list-group-item">A third item</li>
  </ul>
  <div class="card-body">
    <a href="#" class="card-link">Card link</a>
    <a href="#" class="card-link">Another link</a>
  </div>
</div>
```

Header and footer: - set header and footer of card

Classes: - card-header, card-footer

Ex: -

```
<div class="card text-center">
  <div class="card-header">Featured</div>
  <div class="card-body">
    <h5 class="card-title">Special title treatment</h5>
    <p class="card-text">
      With supporting text below as a natural lead-in to additional content
    </p>
    <a href="#" class="btn btn-primary">Go somewhere</a>
  </div>
  <div class="card-footer text-body-secondary">2 days ago</div>
</div>
```

Sizing: - card take 100% width by default or set width to card

Using grid: -

Ex: -

```
<div class="row">
  <div class="col-sm-6 mb-3 mb-sm-0">
    <div class="card"> ... </div>
  </div>
  <div class="col-sm-6">
    <div class="card"> ... </div>
  </div>
</div>
```

Using utilities: - use width utilities to set card width

Ex: -

```
<div class="card w-75"> ... </div>
<div class="card w-50"> ... </div>
```

Using custom CSS: -

Ex: -

```
<div class="card" style="width: 18rem;"> ... </div>
```

Text alignment: -
set text alignment of cards using .text-center, .text-end and .text-start utilities

Ex: -

```
<div class="card text-start" style="width: 18rem;"> ... </div>
<div class="card text-center" style="width: 18rem;"> ... </div>
<div class="card text-end" style="width: 18rem;"> ... </div>
```

Navigation to card: -
set navigation in car-header

Ex: -

```
<div class="card text-center">
   <div class="card-header">
      <ul class="nav nav-tabs card-header-tabs">
         <li class="nav-item">
            <a class="nav-link active" aria-current="true" href="#">Active</a>
         </li>
         <li class="nav-item">
            <a class="nav-link" href="#">Link</a>
         </li>
         <li class="nav-item">
            <a class="nav-link disabled" aria-disabled="true">Disabled</a>
         </li>
      </ul>
   </div>
   <div class="card-body">
      <h5 class="card-title">Special title treatment</h5>
      <p class="card-text">
         With supporting text below as a natural lead-in to additional content
      </p>
      <a href="#" class="btn btn-primary">Go somewhere</a>
   </div>
</div>
```

Image overlays: -
set image in background and overlay text on image

Note: - content should not be larger than image height

Classes: - card-img-overlay

Ex: -

```
<div class="card text-bg-dark m-2" style="width: 22rem;">
   <img src="../3css/res/cityimg.jpg" class="card-img">
   <div class="card-img-overlay">
      <h5 class="card-title">Card title</h5>
```

```
        <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>Last updated 3 mins ago</small></p>
    </div>
</div>
```

Horizontal card: - set horizontal card using combination grid and utility classes

Ex: -

```
<div class="card mb-3 m-2" style="max-width: 540px;">
    <div class="row g-0">
        <div class="col-md-4">
            <img src="../3css/res/cityimg.jpg" class="img-fluid rounded-start" style="height: 100%; object-
fit: cover; object-position: right;">
        </div>
        <div class="col-md-8">
            <div class="card-body">
                <h5 class="card-title">Card title</h5>
                <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-body-secondary">Last updated 3 mins ago</small>
                </p>
            </div>
        </div>
    </div>
</div>
```

Card styles: -

Background and colors: - set background color with text to card

Ex: -

```
<div class="card text-bg-dark mb-3 m-2" style="max-width: 18rem;">
    <div class="card-header">Header</div>
    <div class="card-body">
        <h5 class="card-title">Primary card title</h5>
        <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>
```

Borders: - set border to class with appropriate text color

Ex: -

```
<div class="card border-primary mb-3 m-2" style="max-width: 18rem;">
   <div class="card-header">Header</div>
   <div class="card-body text-primary">
      <h5 class="card-title">Primary card title</h5>
      <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>
```

Mix-ins utilities: - in card can remove background of card header and footer, and also set background transparent

Ex: -

```
<div class="card border-success mb-3" style="max-width: 18rem;">
   <div class="card-header bg-transparent border-success">Header</div>
   <div class="card-body text-success">
      <h5 class="card-title">Success card title</h5>
      <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 class="card-footer bg-transparent border-success">Footer</div>
</div>
```

Card layout: -

Card group: - it renders cards as a single, attached element with equal width and height columns

Class: - card-group

Ex: -

```
<div class="card-group">
   <div class="card"> ... </div>
   <div class="card"> ... </div>
   <div class="card"> ... </div>
</div>
```

Grid cards: - set cards in .row-cols classes to control how many grid columns shows per rows

Ex: -

```
<div class="row row-cols-1 row-cols-md-2 g-4">
   <div class="col">
      <div class="card"> ... </div>
   </div>
   <div class="col">
      <div class="card"> ... </div>
   </div>
   <div class="col">
      <div class="card"> ... </div>
   </div>
   <div class="col">
      <div class="card"> ... </div>
   </div>
</div>
```

Masonry: - it set masonry effects on cards in row, use plug-in for masonry effect, it is not directly include in bootstrap
Script: - <script src="https://cdn.jsdelivr.net/npm/masonry-layout@4.2.2/dist/masonry.pkgd.min.js" integrity="sha384-GNFwBvfVxBkLMJpYMOABq3c+d3KnQxudP/mGPkzpZSTYykLBNsZEnG2D9G/X/+7D" crossorigin="anonymous" async></script>
Links: - https://getbootstrap.com/docs/5.3/examples/masonry/
Setup: - set data-masonry attribute value as '{"percentPosition":true}' in element of .row-cols
Attribute: - data-masonry='{"percentPosition": true}'
Ex: -

```
<div class="row row-cols-2 row-cols-md-4 g-4 m-3" data-masonry='{"percentPosition":true}'>
   <div class="col">
      <div class="card"> ... </div>
   </div>
   <div class="col">
      <div class="card"> ... </div>
   </div>
   <div class="col">
      <div class="card"> ... </div>
   </div>
   <div class="col">
      <div class="card"> ... </div>
   </div>
</div>
```

Close button: - set close button for dismissing content like alerts, offcanvas.
Class: - btn-close,
Ex: -

```
<button type="button" class="btn-close" aria-label="Close"></button>
```

Disabled state: - set disabled attribute to button for disabled it.
Ex: -

```
<button type="button" class="btn-close" disabled aria-label="Close"></button>
```

Off-canvas: - build hidden sidebars into project with some classes and javascript plug-in

Off-canvas components: -

Classes: - offcanvas, offcanvas-start, show, offcanvas-header, offcanvas-title, btn-close, offcanvas-body

- .offcanvas .show: - show content of offcanvas

Attribute: - set attribute data-bs-dismiss="offcanvas" to button with .class-close in offcanvas

Ex: -

```
<div class="offcanvas offcanvas-start show" tabindex="-1">
  <div class="offcanvas-header">
    <h5 class="offcanvas-title" id="offcanvasLabel">Offcanvas</h5>
    <button type="button" class="btn-close" data-bs-dismiss="offcanvas"></button>
  </div>
  <div class="offcanvas-body">
    Content for the offcanvas goes here. You can place just about any Bootstrap component or custom
elements here.
  </div>
</div>
```

Live demo: - show offcanvas on button click, setup data-bs-toggle="offcanvas" at <a> or <button> element. Target id of .offcanvas with href for <a> element or use data-bs-target attribute to target offcanvas id for <button> element.

Classes: - offcanvas

- Offcanvas: - hide content (default)
- offcanvas-start: - place offcanvas at left position

Attribute: - data-bs-toggle="offcanvas", data-bs-target="#oc1"

Ex: -

```
<a class="btn btn-primary" data-bs-toggle="offcanvas" href="#oc1"> Link with href </a>
<button class="btn btn-primary" type="button" data-bs-toggle="offcanvas" data-bs-target="#oc1">
  Button with data-bs-target
</button>

<div class="offcanvas offcanvas-start" tabindex="-1" id="oc1">
  <div class="offcanvas-header">
    <h5 class="offcanvas-title">Offcanvas</h5>
    <button type="button" class="btn-close" data-bs-dismiss="offcanvas"></button>
  </div>
  <div class="offcanvas-body">
    <div>
      Some text as placeholder, in real life you can have the elements you have chosen. Like text,
images, lists, etc.
    </div>
    <div class="dropdown mt-3">
```

```
      <button class="btn btn-secondary dropdown-toggle" type="button" data-bs-
toggle="dropdown">
        Dropdown button
      </button>
      <ul class="dropdown-menu">
        <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>
  </div>
</div>
```

Body scrolling: - use attribute to data-bs-scroll to enable body scrolling while offcanvas is shown. Set that attribute to .offcanvas element, for show backdrop with body scrolling set value of data-bs-backdrop attribute as true, it is default true.

Attribute: - data-bs-scroll="true", data-bs-backdrop="false"

Ex: -

```
<button class="btn btn-primary" type="button" data-bs-toggle="offcanvas" data-bs-target="#oc2">
Enable body scrolling</button>

<div class="offcanvas offcanvas-start" data-bs-scroll="true" data-bs-backdrop="false" tabindex="-1"
id="oc2">
  <div class="offcanvas-header">
    <h5 class="offcanvas-title">Offcanvas with body scrolling</h5>
    <button type="button" class="btn-close" data-bs-dismiss="offcanvas"></button>
  </div>
  <div class="offcanvas-body">
    <p>Try scrolling the rest of the page to see this option in action.</p>
  </div>
</div>
```

Static backdrop: - set static backdrop for close off-canvas when only click on close button.

Attribute: - data-bs-backdrop="static"

Ex: -

```
<button class="btn btn-primary" type="button" data-bs-toggle="offcanvas" data-bs-target="#oc3">
  Toggle static offcanvas
</button>

<div class="offcanvas offcanvas-start" data-bs-backdrop="static" tabindex="-1" id="oc3">
  <div class="offcanvas-header">
    <h5 class="offcanvas-title">Offcanvas</h5>
    <button type="button" class="btn-close" data-bs-dismiss="offcanvas"></button>
  </div>
  <div class="offcanvas-body">
    <div> I will not close if you click outside of me. </div>
  </div>
</div>
```

Dark offcanvas: - set .text-bg-dark to .offcanvas, and .btn-close-white with .btn-close to <button> element, also can set data-bs-theme="dark" to .offcanvas or root element

Classes: - text-bg-dark, btn-close-white

Ex: -

```
<button class="btn btn-primary" type="button" data-bs-toggle="offcanvas" data-bs-target="#oc4">
  Dark offcanvas
</button>

<div class="offcanvas offcanvas-start text-bg-dark" tabindex="-1" id="oc4">
  <div class="offcanvas-header">
    <h5 class="offcanvas-title">Offcanvas</h5>
    <button type="button" class="btn-close btn-close-white" data-bs-dismiss="offcanvas"></button>
  </div>
  <div class="offcanvas-body">
    <p>Place offcanvas content here. </p>
  </div>
</div>
```

Ex: - data-bs-theme="dark"

```
<div class="offcanvas offcanvas-start" data-bs-theme="dark" tabindex="-1" id="oc4">
```

Responsive: - set offcanvas with breakpoint, hide content in offcanvas from a specified breakpoint and down, about the breakpoint contents will behave as usual,

Classes: - offcanvas, offcanvas-sm, offcanvas-md, offcanvas-lg, offcanvas-xl, offcanvas-xxl

Note: - set data-bs-target attribute in close button to target .offcanvas container id for close button working properly.

Ex: -

```
<button class="btn btn-primary d-lg-none" type="button" data-bs-toggle="offcanvas" data-bs-target="#oc5"> Toggle offcanvas </button>

<div class="alert alert-info d-none d-lg-block">
  Resize browser to show the responsive offcanvas toggle.
</div>
<div class="offcanvas-lg offcanvas-start" tabindex="-1" id="oc5">
  <div class="offcanvas-header">
    <h5 class="offcanvas-title">Responsive offcanvas</h5>
    <button type="button" class="btn-close" data-bs-dismiss="offcanvas" data-bs-target="#oc5">
</button>
  </div>
  <div class="offcanvas-body">
    <ul class="nav flex-column flex-lg-row">
      <li class="nav-item">
        <a class="nav-link active" aria-current="page" href="#">Active</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Link</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Link</a>
      </li>
      <li class="nav-item">
        <a class="nav-link disabled" aria-disabled="true">Disabled</a>
      </li>
    </ul>
  </div>
</div>
```

Placement: - there is no default placement for off-canvas component, it must be add

Classes: - offcanvas-start, offcanvas-end, offcanvas-top, offcanvas-bottom

Ex: -

```
<button class="btn btn-primary" type="button" data-bs-toggle="offcanvas" data-bs-target="#oc6">
  Toggle top offcanvas
</button>

<div class="offcanvas offcanvas-top" tabindex="-1" id="oc6">
  <div class="offcanvas-header">
    <h5 class="offcanvas-title">Offcanvas top</h5>
    <button type="button" class="btn-close" data-bs-dismiss="offcanvas"></button>
  </div>
  <div class="offcanvas-body">
    Content of offcanvas body
  </div>
</div>
```

Navbar: - it supports branding, navigation

Ex: -

```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar</a>
```

```
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#nav1">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="nav1">
        <ul class="navbar-nav me-auto mb-2 mb-lg-0">
          <li class="nav-item">
            <a class="nav-link active" 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="#" role="button" data-bs-toggle="dropdown">
              Dropdown
            </a>
            <ul class="dropdown-menu">
              <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" role="search">
          <input class="form-control me-2" type="search" placeholder="Search">
          <button class="btn btn-outline-success" type="submit">Search</button>
        </form>
      </div>
  </div>
</nav>
```

Brand: - it set brand name in navbar, it can set image, text or both

Classes: - navbar, navbar-brand

Ex: -

```
<nav class="navbar bg-body-tertiary">
  <div class="container-fluid">
    <a class="navbar-brand mb-0 h1" href="#">
      <img src="../3css/res/cityimg.jpg" alt="Logo" width="30px" height="24px" class="d-inline-block
align-text-top"> Navbar
    </a>
  </div>
```

```
</nav>
```

Nav: - set nav in navbar after .navbar-brand
Class: - navbar-nav
Ex: -

```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#nav2">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="nav2">
      <ul class="navbar-nav">
        <li class="nav-item">
          <a class="nav-link active" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Features</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Pricing</a>
        </li>
        <li class="nav-item">
          <a class="nav-link disabled">Disabled</a>
        </li>
      </ul>
    </div>
  </div>
</nav>
```

Forms in navbar: - place form controls in navbar
Ex: -

```
<nav class="navbar bg-body-tertiary">
  <div class="container-fluid">
    <a class="navbar-brand">Navbar</a>
    <form class="d-flex" role="search">
      <input class="form-control me-2" type="search" placeholder="Search">
      <button class="btn btn-outline-success" type="submit">Search</button>
    </form>
  </div>
</nav>
```

Text in navbar: - set text in navbar

Class: - navbar-text

Ex: -

```
<nav class="navbar bg-body-tertiary">
   <div class="container-fluid">
      <span class="navbar-text"> Navbar text with an inline element </span>
   </div>
</nav>
```

Color schemes: - set color to navbar, or use data-bs-theme="dark" for dark navbar

Attribute: - data-bs-theme="dark"

Ex: - set dark theme to navbar

```
<nav class="navbar navbar-expand-lg bg-body-tertiary" data-bs-theme="dark">
   <!-- Navbar content -->
</nav>
```

Ex: - set background with dark theme and background color utilities

```
<nav class="navbar navbar-expand-lg bg-success" data-bs-theme="dark">
   <!-- Navbar content -->
</nav>
```

Ex: - set background color in style attribute with background-color property

```
<nav class="navbar navbar-expand-lg" style="background-color: #e3f2fd;">
   <!-- Navbar content -->
</nav>
```

Container: - take navbar to center of page by wrapping it in .container

Ex: -

```
<div class="container">
   <nav class="navbar navbar-expand-lg bg-success">
      <div class="container-fluid">
         <a class="navbar-brand" href="#">Navbar</a>
      </div>
   </nav>
</div>
```

Placement: - set position of navbar

Classes: - fixed-top, fixed-bottom, sticky-top, sticky-bottom

Ex: -

```
<nav class="navbar fixed-top bg-body-tertiary">
   <div class="container-fluid">
      <a class="navbar-brand" href="#">Fixed top</a>
   </div>
</nav>
```

Scrolling: - set vertical scrolling to toggle-able collapsed navbar, also can use --bs-scroll-height variable for change default scroll height

Class: - navbar-nav-scroll

Attribute: - --bs-scroll-height: 100px; //default(75vh)

Ex: -

```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar scroll</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#nav4">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="nav4">
      <ul class="navbar-nav me-auto my-2 my-lg-0 navbar-nav-scroll" style="--bs-scroll-height:
100px;">
        <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="#">Link1</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Link2</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Link3</a>
        </li>
        <li class="nav-item dropdown">
          <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown">
            Dropdown
          </a>
          <ul class="dropdown-menu">
            <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 Link</a>
        </li>
      </ul>
      <form class="d-flex" role="search">
        <input class="form-control me-2" type="search" placeholder="Search">
```

```
            <button class="btn btn-outline-success" type="submit">Search</button>
         </form>
      </div>
   </div>
</nav>
```

Responsive behaviors: - Navbars can use .navbar-toggler, .navbar-collapse, and .navbar-expand {-sm|-md|-lg|-xl|-xxl} classes to determine when their content collapses behind a button.
Classes: - .navbar-expand-sm, .navbar-expand-md, .navbar-expand-lg, .navbar-expand-xl, .navbar-expand-xxl

Toggler: - Navbar togglers are left-aligned by default, but they follow sibling element as .navbar-brand
Setup: - set .navbar-expand-lg to .navbar for expanding navbar on given breakpoint, set button element with navbar-toggler in .container of .navbar, also set .navbar-toggler-icon to span element in button element, on button element set attribute as data-bs-toggle="collapse" and data-bs-target to target id of .navbar-nav's container. Also set .collapse and .navbar-collapse to navbar-nav's container, for collapse navbar menu when screen size less than breakpoint that assign to .navbar container with navbar-expand-breakpoint.
Classes: - navbar-toggler, navbar-toggler-icon, collapse, navbar-collapse, navbar-nav
Attribute: - data-bs-toggle="collapse", data-bs-target="#target"

Ex: - add .navbar-brand element with navbar-nav for hide navbar-brand name when navbar is collapse, it will set navbar-toggler button to left side.

```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
   <div class="container-fluid">
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#nav5">
         <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="nav5">
         <a class="navbar-brand" href="#">Hidden brand</a>
         <ul class="navbar-nav me-auto mb-2 mb-lg-0">
            <li class="nav-item">
               <a class="nav-link active" href="#">Home</a>
            </li>
            <li class="nav-item">
               <a class="nav-link" href="#">Link</a>
            </li>
            <li class="nav-item">
               <a class="nav-link disabled">Disabled</a>
            </li>
         </ul>
```

```
            <form class="d-flex" role="search">
                <input class="form-control me-2" type="search" placeholder="Search">
                <button class="btn btn-outline-success" type="submit">Search</button>
            </form>
        </div>
    </div>
</nav>
```

Ex: - set .navbar-brand element before .navbar-toggler button element for show brand name at left side and toggle button to far-right of navbar

```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
    <div class="container-fluid">
        <a class="navbar-brand" href="#">brand name</a>
        <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#nav5">
            <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="nav5">
            ... ... ...
        </div>
    </div>
</nav>
```

Ex: - set .navbar-brand element after .navbar-toggler button element for show brand name at right side and toggle button to far-left of navbar

```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
    <div class="container-fluid">
        <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#nav5">
            <span class="navbar-toggler-icon"></span>
        </button>
        <a class="navbar-brand" href="#">brand name</a>
        <div class="collapse navbar-collapse" id="nav5">
            ... ... ...
        </div>
    </div>
</nav>
```

External content: - show content that outside the navbar, when click on navbar toggle button
Setup: - target id of external .collapse container with navbar’s toggle button using data-bs-target attribute, also set data-bs-toggle="collapse" attribute to that button
Ex: -

```
<div class="collapse" id="nav6" data-bs-theme="dark">
    <div class="bg-dark p-4">
```

```
      <h5 class="text-body-emphasis h4">Collapsed content</h5>
      <span class="text-body-secondary">Toggleable via the navbar brand.</span>
   </div>
</div>
<nav class="navbar navbar-dark bg-dark">
   <div class="container-fluid">
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#nav6">
         <span class="navbar-toggler-icon"></span>
      </button>
   </div>
</nav>
```

Offcanvas navbar: -

Ex: -

```
<nav class="navbar navbar-expand-lg bg-body-tertiary fixed-top">
   <div class="container-fluid">
      <a class="navbar-brand" href="#">Offcanvas navbar</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="offcanvas" data-bs-target="#nv7">
         <span class="navbar-toggler-icon"></span>
      </button>
      <div class="offcanvas offcanvas-end" tabindex="-1" id="nv7">
         <div class="offcanvas-header">
            <h5 class="offcanvas-title">Offcanvas</h5>
            <button type="button" class="btn-close" data-bs-dismiss="offcanvas"></button>
         </div>
         <div class="offcanvas-body">
            <ul class="navbar-nav justify-content-end flex-grow-1 pe-3">
               <li class="nav-item">
                  <a class="nav-link active" 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="#" role="button" data-bs-toggle="dropdown"
> Dropdown </a>
                  <ul class="dropdown-menu">
                     <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>
            </ul>
            <form class="d-flex mt-3" role="search">
               <input class="form-control me-2" type="search" placeholder="Search">
               <button class="btn btn-outline-success" type="submit">Search</button>
            </form>
         </div>
      </div>
   </div>
</nav>
```

Placeholders: - use it for indicate that something is still loading

Classes: - placeholder, placeholder-glow

Ex: -

```
<div class="card m-3" style="width: 18rem;">
   <img src="../3css/res/background.jpg" class="card-img-top">
   <div class="card-body">
      <h5 class="card-title placeholder-glow">
         <span class="placeholder col-6"></span>
      </h5>
      <p class="card-text placeholder-glow">
         <span class="placeholder col-7"></span>
         <span class="placeholder col-4"></span>
         <span class="placeholder col-4"></span>
         <span class="placeholder col-6"></span>
         <span class="placeholder col-8"></span>
      </p>
      <a class="btn btn-primary disabled placeholder col-6" aria-disabled="true"></a>
   </div>
</div>
```

Width: - set width of placeholder using grid column classes, width utilities or inline style.

Ex: -

```
<span class="placeholder col-6"></span>
<span class="placeholder w-75"></span>
<span class="placeholder" style="width: 25%;"></span>
```

Background color: - set background color with color utilities

Ex: -

```
<span class="placeholder col-12 bg-primary"></span>
```

Sizing: - set size placeholder using modifiers

Classes: - placeholder-lg, placeholder-sm, placeholder-xs

Ex: -

```
<span class="placeholder col-12 placeholder-lg"></span>
```

```
<span class="placeholder col-12"></span>
<span class="placeholder col-12 placeholder-sm"></span>
<span class="placeholder col-12 placeholder-xs"></span>
```

Animation: - set placeholder’s container with classes for glow or wave animation

Classes: - placeholder-glow, placeholder-wave

Ex: -

```
<p class="placeholder-glow">
   <span class="placeholder bg-primary col-12"></span>
</p>
<p class="placeholder-wave">
   <span class="placeholder bg-danger col-12"></span>
</p>
```

Progress: - it set custom style of bootstrap to progress bar, set .progress as wrapper and indicate max value, use .progress-bar to show progress, set it, in .progress container.

Classes: - progress, progress-bar

Ex: -

```
<div class="progress">
   <div class="progress-bar" style="width: 25%"></div>
</div>
```

Bar sizing: -

Width: - use utilities for set bar width

Ex: -

```
<div class="progress">
   <div class="progress-bar w-50"></div>
</div>
```

Height: - set height value to .progress container, inner .progress-bar value will change accordingly

Ex: -

```
<div class="progress" style="height: 30px;">
   <div class="progress-bar" style="width: 25%"></div>
</div>
```

Labels: - set text in .progress bar

Ex: -

```
<div class="progress">
   <div class="progress-bar" style="width: 75%">75%</div>
</div>
```

Ex: - set .overflow-visble to show text outside of progress bar, also set .text-dark

```
<div class="progress">
   <div class="progress-bar overflow-visible text-dark" style="width: 10%">
```

```
    set .overflow-visble to show text outside of progress bar
  </div>
</div>
```

Background: - use background utilities to set background color

Ex: -

```
<div class="progress">
  <div class="progress-bar bg-info" style="width: 30%"></div>
</div>
```

Multiple bars: - include multiple progress-bars using .progress-stacked to create a single stacked progress bar. Width must be applied to .progress container rather than .progress-bar

Class: - progress-stacked

Ex: -

```
<div class="progress-stacked">
  <div class="progress" style="width: 13%">
    <div class="progress-bar bg-success" ></div>
  </div>
  <div class="progress" style="width: 23%">
    <div class="progress-bar bg-danger"></div>
  </div>
  <div class="progress" style="width: 33%">
    <div class="progress-bar bg-warning"></div>
  </div>
</div>
```

Striped: - set progress bar with striped

Class: - progress-bar-striped

Ex: -

```
<div class="progress">
  <div class="progress-bar progress-bar-striped bg-danger" style="width: 33%"></div>
</div>
```

Animated bar: - set animation to striped progress bar

Class: -

Ex: -

```
<div class="progress">
  <div class="progress-bar progress-bar-striped progress-bar-animated bg-danger" style="width: 33%"
></div>
</div>
```

Spinners: - it indicate the loading state of component or bootstrap

Border spinner: - use it for lightweight loading indicator

Ex: -

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

Spinner colors: - use text color utilities to change color of spinner

Classes: - text-primary, text-danger, text-warning, etc.

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

Growing spinner: - it grows repeatedly, use text color utilities to change color

Classes: - spinner-grow

Ex: -

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

Alignment margin: - use margin utilities for alignment

Ex: -

```
<div class="spinner-grow text-info m-4"></div>
```

Placement: - place spinner using flex, float, and text-align utilities

Flex: - use flex utilities for placement of spinner

Ex: -

```
<div class="d-flex justify-content-center">
   <div class="spinner-border" role="status"></div>
</div>
```

Ex: - spinner with name

```
<div class="d-flex align-items-center container-fluid">
   <strong role="status">Loading...</strong>
   <div class="spinner-border ms-auto"></div>
</div>
```

Float: - use float utilities for placement

Ex: -

```
<div class="clearfix container-fluid">
   <div class="spinner-border float-end"></div>
</div>
```

Text-align: - use text-align utilities for placement

Ex: -

```
<div class="text-center">
   <div class="spinner-border" role="status"></div>
</div>
```

Spinner size: - set size of spinner using custom CSS or use spinner-border-sm or spinner-grow-sm classes for small spinners

Classes: - spinner-border-sm, spinner-grow-sm

Ex: -

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

Button spinners: - use spinners inside buttons

Ex: -

```
<button class="btn btn-primary" type="button" disabled>
   <span class="spinner-border spinner-border-sm" aria-hidden="true"></span>
   <span role="status">Loading...</span>
</button>

<button class="btn btn-success" type="button" disabled>
   <span class="spinner-grow spinner-grow-sm" aria-hidden="true"></span>
   <span role="status">Loading...</span>
</button>
```

Accordion: - build vertically collapsing accordions

Setup: - set container .accordion, also set id to that container, in that container set .accordion-item container, in that container set .accordion-header, in header set accordion-button, set attributes data-bs-toggle="collapse" and data-bs-target to that button, and target container .accordion-body's container id, that will set follow by button, set container of .accordion-body with id that targeted by button, also set .accordion-collapse, .collapse to container, also set .show to that container for show it open, also set attribute data-bs-parent to target .accordion container's id, in that container set element with .accordion-body that will show content in accordion.

To show accordion item is not open set .collapsed to button, do not add .show utility to container of .accordion-body

Classes: - accordion, accordion-item, accordion-header, accordion-button, collapsed, accordion-collapse, collapse, show, accordion-body

Attribute: - data-bs-toggle="collapse", data-bs-target, data-bs-parent

Ex: -

```
<div class="accordion m-2" id="ac1">
   <div class="accordion-item">
      <h2 class="accordion-header">
         <button class="accordion-button" type="button" data-bs-toggle="collapse" data-bs-target=
"#ac1c1"> Accordion Item #1 </button>
      </h2>
      <div id="ac1c1" class="accordion-collapse collapse show" data-bs-parent="#ac1">
         <div class="accordion-body">
            <strong>This is the first item's accordion body.</strong> It is shown by default, until the
collapse plugin adds the appropriate classes that we use to style each element. These classes control
the overall appearance, as well as the showing and hiding via CSS transitions. You can modify any of
this with custom CSS or overriding our default variables. It's also worth noting that just about any
HTML can go within the <code>.accordion-body</code>, though the transition does limit overflow.
```

```
      </div>
    </div>
  </div>
  <div class="accordion-item">
    <h2 class="accordion-header">
      <button class="accordion-button collapsed" type="button" data-bs-toggle="collapse" data-bs-
target="#ac1c2"> Accordion Item #2 </button>
    </h2>
    <div id="ac1c2" class="accordion-collapse collapse" data-bs-parent="#ac1">
      <div class="accordion-body">
        <strong>This is the second item's accordion body.</strong> It is hidden by default, until the
collapse plugin adds the appropriate classes that we use to style each element. These classes control
the overall appearance, as well as the showing and hiding via CSS transitions. You can modify any of
this with custom CSS or overriding our default variables. It's also worth noting that just about any
HTML can go within the <code>.accordion-body</code>, though the transition does limit overflow.
      </div>
    </div>
  </div>
  <div class="accordion-item">
    <h2 class="accordion-header">
      <button class="accordion-button collapsed" type="button" data-bs-toggle="collapse" data-bs-
target="#ac1c3"> Accordion Item #3 </button>
    </h2>
    <div id="ac1c3" class="accordion-collapse collapse" data-bs-parent="#ac1">
      <div class="accordion-body">
        <strong>This is the third item's accordion body.</strong> It is hidden by default, until the
collapse plugin adds the appropriate classes that we use to style each element. These classes control
the overall appearance, as well as the showing and hiding via CSS transitions. You can modify any of
this with custom CSS or overriding our default variables. It's also worth noting that just about any
HTML can go within the <code>.accordion-body</code>, though the transition does limit overflow.
      </div>
    </div>
  </div>
</div>
```

Flush: - set it for remove some borders and rounded corners to set as edge-to-edge

Class: - accordion-flush

Ex: -

```
<div class="accordion m-2 accordion-flush" id="ac1">
  ... ... ...
</div>
```

Always open: - remove data-bs-parent attribute from each .accordion-collapse, to make accordion items stay open when another accordion item is opened

Ex: -

```
<div class="accordion m-2" id="ac2">
  <div class="accordion-item">
    <h2 class="accordion-header">
      <button class="accordion-button collapsed" type="button" data-bs-toggle="collapse" data-bs-target="#ac2c1"> Accordion Item #1 </button>
    </h2>
    <div id="ac2c1" class="accordion-collapse collapse">
      <div class="accordion-body">
        <strong>This is the first item's accordion body.</strong> It is shown by default, until the collapse plugin adds the appropriate classes that we use to style each element. These classes control the overall appearance, as well as the showing and hiding via CSS transitions. You can modify any of this with custom CSS or overriding our default variables. It's also worth noting that just about any HTML can go within the <code>.accordion-body</code>, though the transition does limit overflow.
      </div>
    </div>
  </div>
  <div class="accordion-item">
    <h2 class="accordion-header">
      <button class="accordion-button collapsed" type="button" data-bs-toggle="collapse" data-bs-target="#ac2c2"> Accordion Item #2 </button>
    </h2>
    <div id="ac2c2" class="accordion-collapse collapse">
      <div class="accordion-body">
        <strong>This is the second item's accordion body.</strong> It is hidden by default, until the collapse plugin adds the appropriate classes that we use to style each element. These classes control the overall appearance, as well as the showing and hiding via CSS transitions. You can modify any of this with custom CSS or overriding our default variables. It's also worth noting that just about any HTML can go within the <code>.accordion-body</code>, though the transition does limit overflow.
      </div>
    </div>
  </div>
</div>
```

Pagination: - it indicates a series of related content exists across multiple pages.

Classes: - pagination, page-item, page-link

Ex: -

```
<nav aria-label="Page navigation example">
  <ul class="pagination">
    <li class="page-item"><a class="page-link" href="#">Previous</a></li>
    <li class="page-item"><a class="page-link" href="#">1</a></li>
```

```
        <li class="page-item"><a class="page-link" href="#">2</a></li>
        <li class="page-item"><a class="page-link" href="#">3</a></li>
        <li class="page-item"><a class="page-link" href="#">Next</a></li>
    </ul>
</nav>
```

Working with icons: - set &laquo; and &raquo; icons in pagination

Ex: -

```
<nav aria-label="Page navigation example">
    <ul class="pagination">
        <li class="page-item">
            <a class="page-link" href="#" aria-label="Previous">
                <span aria-hidden="true">&laquo;</span>
            </a>
        </li>
        <li class="page-item"><a class="page-link" href="#">1</a></li>
        <li class="page-item"><a class="page-link" href="#">2</a></li>
        <li class="page-item"><a class="page-link" href="#">3</a></li>
        <li class="page-item">
            <a class="page-link" href="#" aria-label="Next">
                <span aria-hidden="true">&raquo;</span>
            </a>
        </li>
    </ul>
</nav>
```

Disabled and active state: - set .page-item as active or disabled

Classes: - disabled, active

Ex: -

```
<nav>
    <ul class="pagination">
        <li class="page-item disabled"> <a class="page-link">Previous</a> </li>
        <li class="page-item"> <a class="page-link" href="#">1</a></li>
        <li class="page-item active" aria-current="page"> <a class="page-link" href="#">2</a> </li>
        <li class="page-item"> <a class="page-link" href="#">3</a></li>
        <li class="page-item"> <a class="page-link" href="#">Next</a> </li>
    </ul>
</nav>
```

Sizing: - set size to pagination

Classes: - pagination-lg, pagination-sm

Ex: -

```
<nav>
```

```
  <ul class="pagination pagination-lg">
    <li class="page-item active" aria-current="page">
      <span class="page-link">1</span>
    </li>
    <li class="page-item"><a class="page-link" href="#">2</a></li>
    <li class="page-item"><a class="page-link" href="#">3</a></li>
  </ul>
</nav>
```

Alignment: - change alignment of pagination with flex-box utilities

Ex: -

```
<nav aria-label="Page navigation example">
  <ul class="pagination justify-content-center">
    <li class="page-item disabled"><a class="page-link">Previous</a></li>
    <li class="page-item"><a class="page-link" href="#">1</a></li>
    <li class="page-item"><a class="page-link" href="#">2</a></li>
    <li class="page-item"><a class="page-link" href="#">3</a></li>
    <li class="page-item"><a class="page-link" href="#">Next</a></li>
  </ul>
</nav>
```

Carousel: - a slideshow component for cycling through elements-images or slide of text

Setup: - set .carousel wrapper element with .slide and id, in that container set .carousel-inner container, in that container set .carousel-item container, in that set content to display, also set .active to any .carousel-item container, after .carousel-inner container set button with .carousel-control-prev class and data-bs-slide="prev" attribute, also set data-bs-target to target .carousel container id, In that button set span with .carousel-control-prev-icon class for set icon in button
After that button also set another button with .carousel-control-next class and data-bs-slide="next" attribute, also set data-bs-target to target .carousel container's id, In that button set span with .carousel-control-next-icon class for set icon in button

Classes: - carousel, slide, carousel-inner, carousel-item, active, carousel-control-prev, carousel-control-prev-icon, carousel-control-next, carousel-control-next-icon
Attribute: - data-bs-target, data-bs-slide="prev", data-bs-slide="next"

Ex: -

```
<div id="c1" class="carousel slide container">
  <div class="carousel-inner" style="aspect-ratio: 18/9;">
    <div class="carousel-item active h-100">
      <img src="../3css/res/background.jpg" class="d-block w-100 h-100">
    </div>
    <div class="carousel-item h-100">
      <img src="../3css/res/cityimg.jpg" class="d-block w-100 h-100">
```

```
        </div>
        <div class="carousel-item h-100">
            <img src="../3css/res/earthimg.jpg" class="d-block w-100 h-100">
        </div>
    </div>
    <button class="carousel-control-prev" type="button" data-bs-target="#c1" data-bs-slide="prev">
        <span class="carousel-control-prev-icon"></span>
    </button>
    <button class="carousel-control-next" type="button" data-bs-target="#c1" data-bs-slide="next">
        <span class="carousel-control-next-icon"></span>
    </button>
</div>
```

Indicators: - it can add with next and previous buttons, it helps to jump on particular slide
Setup: - setup .carousel-indicators container in .carousel container before .carousel-inner container, in .carousel-indicators set buttons with data-bs-target attribute to target .carousel container's id and data-bs-slide-to value from 0. Also set .active class to button with data-bs-slide-to="0" attribute and value
Class: - carousel-indicators, active
Attributes: - data-bs-target, data-bs-slide-to
Ex: -

```
<div id="c2" class="carousel slide container">
    <div class="carousel-indicators">
        <button type="button" data-bs-target="#c2" data-bs-slide-to="0" class="active"></button>
        <button type="button" data-bs-target="#c2" data-bs-slide-to="1"></button>
        <button type="button" data-bs-target="#c2" data-bs-slide-to="2"></button>
    </div>
    <div class="carousel-inner" style="aspect-ratio: 18/9;"> ... </div>
    ... ... ...
</div>
```

Caption: - set caption of slide inside .carousel-item container with .carousel-caption container, also set .d-none and .d-md-block to .carousel-caption container for show caption on medium-sized devices and above
Classes: - carousel-caption, d-none, d-md-block
Ex: -

```
<div id="c3" class="carousel slide container">
    <div class="carousel-indicators"> ... </div>
    <div class="carousel-inner" style="aspect-ratio: 18/9;">
        <div class="carousel-item active h-100">
            <img src="../3css/res/background.jpg" class="d-block w-100 h-100">
            <div class="carousel-caption d-none d-md-block">
                <h5>Butterfly</h5>
            <p>Blue butterfies of hovering on blue flowers </p>
         </div>
      </div>
      <div class="carousel-item h-100">
         <img src="../3css/res/cityimg.jpg" class="d-block w-100 h-100">
         <div class="carousel-caption d-none d-md-block">
            <h5>City</h5>
            <p>City dazzling at night time </p>
         </div>
      </div>
      <div class="carousel-item h-100">
         <img src="../3css/res/earthimg.jpg" class="d-block w-100 h-100">
         <div class="carousel-caption d-none d-md-block">
            <h5>Earth</h5>
            <p>Green earth </p>
         </div>
      </div>
   </div>
... ... ...
</div>
```

Cross-fade: - set fade transition to carousel component
Class: - carousel-fade
Ex: -

```
<div id="c3" class="carousel slide container carousel-fade"> ... </div>
```

Auto-playing carousels: - set carousel to auto-play with data-bs-ride set as carousel in .carousel container, auto-play will pause when mouse hovered on carousel.
Attribute: - data-bs-ride="carousel"
Ex: -

```
<div id="c3" class="carousel slide container carousel-fade" data-bs-ride="carousel"> ... </div>
```

Auto-playing on user interaction: - set data-bs-ride value as true for start auto-play on user interaction
Ex: -

```
<div id="c3" class="carousel slide container carousel-fade" data-bs-ride="true"> ... </div>
```

Individual interval: - it changes the amount of time to delay between automatically cycling to the next item. Set data-bs-interval to each .carousel-item
Attribute: - data-bs-interval="1000" time in milliseconds
Ex: -

```
<div id="c3" class="carousel slide container carousel-fade" data-bs-ride="carousel">
   <div class="carousel-indicators"> ... </div>
```

```
  <div class="carousel-inner" style="aspect-ratio: 18/9;">
    <div class="carousel-item active h-100" data-bs-interval="1000">
      <img src="../3css/res/background.jpg" class="d-block w-100 h-100">
      <div class="carousel-caption d-none d-md-block">
        <h5>Butterfly</h5>
        <p>Blue butterfies of hovering on blue flowers </p>
      </div>
    </div>
    <div class="carousel-item h-100" data-bs-interval="500">
      <img src="../3css/res/cityimg.jpg" class="d-block w-100 h-100">
      <div class="carousel-caption d-none d-md-block">
        <h5>City</h5>
        <p>City dazzling at night time </p>
      </div>
    </div>
    <div class="carousel-item h-100" data-bs-interval="3000">
      <img src="../3css/res/earthimg.jpg" class="d-block w-100 h-100">
      <div class="carousel-caption d-none d-md-block">
        <h5>Earth</h5>
        <p>Green earth </p>
      </div>
    </div>
  </div>
  ... ... ...
</div>
```

Auto-play carousels without controls: - it is carousel with slides only.

Ex: -

```
<div id="c3" class="carousel slide container carousel-fade" data-bs-ride="carousel">
  <div class="carousel-inner" style="aspect-ratio: 18/9;">
    <div class="carousel-item active h-100" data-bs-interval="1000">
      <img src="../3css/res/background.jpg" class="d-block w-100 h-100">
      <div class="carousel-caption d-none d-md-block">
        <h5>Butterfly</h5>
        <p>Blue butterfies of hovering on blue flowers </p>
      </div>
    </div>
    <div class="carousel-item h-100" data-bs-interval="500">
      <img src="../3css/res/cityimg.jpg" class="d-block w-100 h-100">
      <div class="carousel-caption d-none d-md-block">
        <h5>City</h5>
        <p>City dazzling at night time </p>
      </div>
    </div>
```

```
        <div class="carousel-item h-100" data-bs-interval="3000">
            <img src="../3css/res/earthimg.jpg" class="d-block w-100 h-100">
            <div class="carousel-caption d-none d-md-block">
                <h5>Earth</h5>
                <p>Green earth </p>
            </div>
        </div>
    </div>
</div>
```

Disable touch swiping: - disable swiping left/right on touch-screen devices to move slides

Attribute: - data-bs-touch="false"

Ex: -

```
<div id="c3" class="carousel slide container carousel-fade" data-bs-touch="false">
    ... ... ...
</div>
```

Dark variant: - set data-bs-theme="dark" in .carousel container for dark variant of carousel

Ex: -

```
<div id="c3" class="carousel slide container carousel-fade" data-bs-theme="dark">
    ... ... ...
</div>
```

Scrollspy: - display corresponding content in container, when click on link

Scroll-spy with navbar: -

Setup: - set navbar, and in .navlink set href to target particular content's id that from scroll-spy container after navbar, set data-bs-spy="scroll" attribute to scroll-spy container with data-bs-target that target id for associative navbar, also set data-bs-smooth-scroll="true" for smooth scrolling, set tabindex="0" for ensure keyboard access, also set custom height and overflow-y: scroll to that container. In container set elements with content and id that will target by navlink

Attributes: - data-bs-spy="scroll", data-bs-target, data-bs-smooth-scroll="true", tabindex="0"

Ex: -

```
<nav id="nb2" class="navbar bg-body-tertiary px-3 mb-3">
    <a class="navbar-brand" href="#">Navbar</a>
    <ul class="nav nav-pills">
        <li class="nav-item"> <a class="nav-link" href="#scroll1">First</a> </li>
        <li class="nav-item"> <a class="nav-link" href="#scroll2">Second</a> </li>
        <li class="nav-item dropdown">
            <a class="nav-link dropdown-toggle" data-bs-toggle="dropdown" href="#">Dropdown</a>
            <ul class="dropdown-menu">
                <li><a class="dropdown-item" href="#scroll3">Third</a></li>
                <li><a class="dropdown-item" href="#scroll4">Fourth</a></li>
                <li><hr class="dropdown-divider"></li>
            <li><a class="dropdown-item" href="#scroll5">Fifth</a></li>
          </ul>
        </li>
    </ul>
</nav>
<div data-bs-spy="scroll" data-bs-target="#nb2" data-bs-smooth-scroll="true" class="bg-body-tertiary
p-3 rounded-2" tabindex="0" style="height: 200px; overflow-y: scroll ;">
    <h4 id="scroll1">First heading</h4>
    <p> ... </p>
    <h4 id="scroll2">Second heading</h4>
    <p> ... </p>
    <h4 id="scroll3">Third heading</h4>
    <p> ... </p>
    <h4 id="scroll4">Fourth heading</h4>
    <p> ... </p>
    <h4 id="scroll5">Fifth heading</h4>
    <p> ... </p>
</div>
```

Nested nav with scroll-spy: - set nested nav with scroll-spy, set custom height and overflow-y properly

Ex: -

```
<div class="row g-0" style="height: 320px;">
    <div class="col-4 h-100">
        <nav id="nv3" class="h-100 flex-column align-items-stretch pe-4 border-end">
            <nav class="nav nav-pills flex-column">
                <a class="nav-link" href="#item-1">Item 1</a>
                <nav class="nav nav-pills flex-column">
                    <a class="nav-link ms-3 my-1" href="#item-1-1">Item 1-1</a>
                    <a class="nav-link ms-3 my-1" href="#item-1-2">Item 1-2</a>
                </nav>
                <a class="nav-link" href="#item-2">Item 2</a>
                <a class="nav-link" href="#item-3">Item 3</a>
                <nav class="nav nav-pills flex-column">
                    <a class="nav-link ms-3 my-1" href="#item-3-1">Item 3-1</a>
                    <a class="nav-link ms-3 my-1" href="#item-3-2">Item 3-2</a>
                </nav>
            </nav>
        </nav>
    </div>
    <div class="col-8 h-100">
        <div data-bs-spy="scroll" data-bs-target="#nv3" data-bs-smooth-scroll="true" tabindex="0"
class="h-100" style="overflow-y: scroll;">
            <div id="item-1">   <h4>Item 1</h4>   <p>...</p> </div>
```

```
      <div id="item-1-1"> <h5>Item 1-1</h5> <p>...</p> </div>
      <div id="item-1-2"> <h5>Item 1-2</h5> <p>...</p> </div>
      <div id="item-2">   <h4>Item 2</h4>   <p>...</p> </div>
      <div id="item-3">   <h4>Item 3</h4>   <p>...</p> </div>
      <div id="item-3-1"> <h5>Item 3-1</h5> <p>...</p> </div>
      <div id="item-3-2"> <h5>Item 3-2</h5> <p>...</p> </div>
    </div>
  </div>
</div>
```

List group with scroll-spy: -

Ex: -

```
<div class="row g-0" style="height: 200px;">
  <div class="col-4 h-100">
    <div id="lst4" class="list-group">
      <a class="list-group-item list-group-item-action" href="#li1">Item 1</a>
      <a class="list-group-item list-group-item-action" href="#li2">Item 2</a>
      <a class="list-group-item list-group-item-action" href="#li3">Item 3</a>
      <a class="list-group-item list-group-item-action" href="#li4">Item 4</a>
    </div>
  </div>
  <div class="col-8 h-100">
    <div data-bs-spy="scroll" data-bs-target="#lst4" data-bs-smooth-scroll="true" tabindex="0"
class="h-100" style="overflow-y: scroll;">
      <h4 id="li1">Item 1</h4> <p>...</p>
      <h4 id="li2">Item 2</h4> <p>...</p>
      <h4 id="li3">Item 3</h4> <p>...</p>
      <h4 id="li4">Item 4</h4> <p>...</p>
    </div>
  </div>
</div>
```

Simple anchor tags: -

Ex: -

```
<div class="row g-0 mb-4" style="height: 200px;">
  <div class="col-4 h-100">
    <div id="ali5" class="d-flex flex-column gap-2 text-center">
      <a class="p-1 rounded" href="#sli-1">Item 1</a>
      <a class="p-1 rounded" href="#sli-2">Item 2</a>
      <a class="p-1 rounded" href="#sli-3">Item 3</a>
      <a class="p-1 rounded" href="#sli-4">Item 4</a>
      <a class="p-1 rounded" href="#sli-5">Item 5</a>
    </div>
  </div>
  <div class="col-8 h-100">
     <div data-bs-spy="scroll" data-bs-target="#ali5" data-bs-smooth-scroll="true" tabindex="0"
class="h-100" style="overflow-y: scroll;">
        <h4 id="sli-1">Item 1</h4> <p>...</p>
        <h4 id="sli-2">Item 2</h4> <p>...</p>
        <h4 id="sli-3">Item 3</h4> <p>...</p>
        <h4 id="sli-4">Item 4</h4> <p>...</p>
        <h4 id="sli-5">Item 5</h4> <p>...</p>
     </div>
  </div>
</div>
```

Bootstrap Themes: - bootstrap themes set different style in bootstrap.

Set attribute 'data-bs-theme' as 'dark' or 'light' to change styling of html page. Apply this attribute to html element for change all elements style. Or also can apply particular element for change its theme.

Syntax: - data-bs-theme='dark' / 'light'
Ex: - apply dark theme to html element.
<html data-bs-theme='dark'>
Ex: - apply light theme to particular element.
<div data-bs-theme='light' class='container border text-bg-primary'>GOOD</div>

Download and apply theme file in bootstrap: -

1) goto website https://bootswatch.com/ to find themes
2) then download any theme's css file from available
3) link path of bootstrap.min.css file in html page after linking bootstrap css links

Ex: -
<link href='https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css' rel='stylesheet' integrity='sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH' crossorigin='anonymous' id='lnk'>
<link href='C:\Users\User_name\Downloads\bootstrap.min.css' rel='stylesheet'>

Customize bootstrap themes: -

1) got to https://bootstrap.build/ there click on open the builder
2) customize colors and styling in side-menu and see applied changes
3) then click on export theme and select bootstrap.min.css file then link it to html page after bootstrap link

Note: - also can select free themes and click on customize and make changes or export it

Customize class properties: -

!important: - this CSS keyword overrides all previous properties for specific element in selectors.

Ex: -

```
<style>
.text-bg-primary{
   background-color: orangered !important;
}
.border{
   border-color: black !important;
}
</style>
<div class='container border text-bg-primary'>GOOD</div>
```

Variables for customize: -

Root: - define variables with values of sizes, colors, etc in root pseudo class with -- this symbol, and use them with var() function in CSS.

Also can add these styles in CSS file.

Ex: -

```
<style>
:root{
   --bs-indigo: indianred;
   --bs-color: red;
}
.text-bg-primary{
   background-color: var(--bs-indigo) !important;
   color: var(--bs-color) !important;
}
</style>
<div class='container border text-bg-primary'>GOOD</div>
```

Set custom data-bs-theme attribute: -

Set data-bs-theme attribute selector when value is theme color, and set different variables values or new variables, those will apply when data-bs-theme attribute value will set as define selector.

syntax: - [data-bs-theme=color]{… … …}

Ex: -

```
<html data-bs-theme='green'>
<style>
[data-bs-theme=green]{
   --bs-body-color: ghostwhite;
   --bs-body-bg: green;
}
.btn{
   --bs-btn-bg: greenyellow;
   background-color: var(--bs-btn-bg) !important;
   border: 1px solid black;
}
</style>
<button type='button' class='btn'>GOOD</button>
```

Bootstrap JS: - use bootstrap with javascript, for enabled javascript and bootstrap javascript bundle and popper links in html page.

Checkbox: -

Indeterminate: - it set checkbox as neither on or off; checkbox can set indeterminate using javascript only. Pseudo class indeterminate is used for style indeterminate checkbox

Js: - object(checkbox).indeterminate=true; //false

Ex: -

```
<style>
   input[type='checkbox']:indeterminate{
      background-color: lime !important;
   }
</style>
<div class="form-check">
   <input class="form-check-input" type="checkbox" id="ch2">
   <label class="form-check-label" for="ch2">Checkbox2</label>
</div>
<script>
   document.getElementById("ch2").indeterminate = true;
</script>
```

Tooltips: - set tooltips with CSS and javascript using bootstrap

Enable tooltips: - include popper.min.js library for positioning before bootstrap.js or use bootstrap.bundle.min.js which contain popper

Tooltips must be initialize before they can used, initialize them using data-bs-toggle attribute

JS code for initialize tooltips: -

```
const tooltipTriggerList = document.querySelectorAll('[data-bs-toggle="tooltip"]')
const tooltiplist = [...tooltipTriggerList].map(tooltipTriggerEl => new bootstrap.Tooltip(tooltipTriggerEl))
```

attributes: -

```
data-bs-toggle="tooltip" // set to element for display tooltip on hover
data-bs-title="Default tooltip" //set title for display in tooltip
```

Ex: -

```
<p>
   Placeholder text to demonstrate some
   <a href="#" data-bs-toggle="tooltip" data-bs-title="Default tooltip">inline links</a> with tooltips.
```

```
</p>
```

Custom tooltip: - set custom variables for css in style for tooltip
CSS tooltip variables: -
--bs-tooltip-bg: red;
--bs-tooltip-color: white;
Attribute: - data-bs-custom-class= "custom-tooltip" //set custom class to tooltip
Ex: -

```
<style>
   .custom-tooltip{
      --bs-tooltip-bg: red;
      --bs-tooltip-color: white;
   }
</style>
<button type="button" class="btn btn-secondary" data-bs-toggle="tooltip" data-bs-custom-class=
"custom-tooltip" data-bs-title="custom tooltip"> Custom tooltip </button>
```

Tooltip directions: - set direction tooltips
Attribute: - data-bs-placement="top" //top, right, bottom, left
Ex: -

```
<button type="button" class="btn btn-secondary" data-bs-toggle="tooltip" data-bs-placement="top"
data-bs-title="Tooltip on top"> Tooltip on top </button>
```

Tooltip with custom html: - set custom html in data-bs-title attribute
Ex: -

```
<button type="button" class="btn btn-secondary" data-bs-toggle="tooltip" data-bs-html="true" data-
bs-title="<em>Tooltip</em> <u>with</u> <b>HTML</b>"> Tooltip with HTML </button>
```

Form validation: -

Custom validation: - set novalidate attribute with .needs-validation class to form element for custom validation
Attribute: - novalidate
Classes: - needs-validation, valid-feedback, invalid-feedback, has-validation
needs-validation: - set validation to form
valid, invalid feedback: - set valid or invalid feedback
has-validation: - set border issue with input group
Ex: -

```
<form class="needs-validation" novalidate>
   <div>
      <label for="uname" class="form-label">Username</label>
      <input type="text" id="uname" class="form-control" required>
      <div class="valid-feedback">ok</div>
      <div class="invalid-feedback">not ok</div>
```

```
    </div>
    <div>
        <label for="eid" class="form-label">Email</label>
        <div class="input-group has-validation">
            <span class="input-group-text">@</span>
            <input type="email" id="eid" class="form-control" required>
            <div class="invalid-feedback">not ok</div>
        </div>
    </div>
    <div>
        <label for="pwd" class="form-label">Password</label>
        <input type="password" id="pwd" class="form-control" required>
        <div class="valid-feedback">ok</div>
        <div class="invalid-feedback">not ok</div>
    </div>
    <button type="submit" class="btn btn-primary">Submit</button>
</form>
```

JS: -

```
<script>
    (()=>{
        "use strict"
        const forms = document.querySelectorAll('.needs-validation');

        const f2=Array.from(forms)
        f2.forEach(validform);
        function validform(form)
        {
            form.addEventListener('submit',valid);
            function valid(event)
            {
                if(!form.checkValidity()){
                event.preventDefault();
                event.stopPropagation();
            }
            form.classList.add('was-validated');
        }
    }
    })()
</script>
```

Array.form(forms): - create array of forms with . needs-validation class.

f2.forEach(validform): - traverse every element for apply validform function

form.addEventListener('submit',valid): - set function valid for every element

form.checkValidity(): - return true if input element contain valid data
event.preventDefault(): - it cancel event if, it cancelable
event.stopPropagation(): - stop propagation of the same event from being called
form.classList.add('was-validated'): - set .was-validated class for show validation messages and style

Advanced code: -

```
// Example starter JavaScript for disabling form submissions if there are invalid fields
(() => {
  'use strict'

  // Fetch all the forms we want to apply custom Bootstrap validation styles to
  const forms = document.querySelectorAll('.needs-validation')

  // Loop over them and prevent submission
  Array.from(forms).forEach(form => {
    form.addEventListener('submit', event => {
      if (!form.checkValidity()) {
        event.preventDefault()
        event.stopPropagation()
      }

      form.classList.add('was-validated')
    }, false)
  })
})()
```

Server-side validation: - bootstrap recommend client side validation but for server side validation use .is-valid and .is-invalid class, .invalid-feedback and .valid-feedback also supports with these classes

Note: - write server side JS code for validation
Classes: - is-valid, is-invalid
Ex: -

```
<form class="row g-3">
    <div class="col-md-4">
        <label for="vs1" class="form-label">First name</label>
        <input type="text" class="form-control is-valid" id="vs1" required>
        <div class="valid-feedback">Looks good!</div>
    </div>
    <div class="col-md-4">
        <label for="vs2" class="form-label">Last name</label>
        <input type="text" class="form-control is-valid" id="vs2" required>
        <div class="valid-feedback">Looks good!</div>
    </div>
    <div class="col-md-4">
        <label for="vsu" class="form-label">Username</label>
        <div class="input-group has-validation">
          <span class="input-group-text">@</span>
          <input type="text" class="form-control is-invalid" id="vsu" required>
          <div class="invalid-feedback">Please choose a username.</div>
        </div>
    </div>
    <div class="col-md-6">
        <label for="vs3" class="form-label">City</label>
        <input type="text" class="form-control is-invalid" id="vs3" required>
        <div class="invalid-feedback">Please provide a valid city.</div>
    </div>
    <div class="col-md-3">
        <label for="vs4" class="form-label">State</label>
        <select class="form-select is-invalid" id="vs4" required>
          <option selected disabled value="">Choose...</option>
          <option value="1">one</option>
          <option value="2">two</option>
          <option value="3">three</option>
        </select>
        <div class="invalid-feedback">Please select a valid state.</div>
    </div>
    <div class="col-md-3">
        <label for="vs5" class="form-label">Zip</label>
        <input type="text" class="form-control is-invalid" id="vs5" required>
        <div class="invalid-feedback">Please provide a valid zip.</div>
    </div>
    <div class="col-12">
        <div class="form-check">
          <input class="form-check-input is-invalid" type="checkbox" value="" id="ic3"  required>
          <label class="form-check-label" for="ic3">Agree to terms and conditions</label>
          <div class="invalid-feedback">You must agree before submitting.</div>
        </div>
    </div>
    <div class="col-12">
        <button class="btn btn-primary" type="submit">Submit form</button>
    </div>
</form>
```

Supported elements: - validation style available for form controls and components

- <input> and <textarea> with .form-control
- <select> with .form-select
- .form-check

Ex: -

```
<form class="row g-3">
  <div class="col-md-4">
    <label for="vs1" class="form-label">First name</label>
    <input type="text" class="form-control is-valid" id="vs1" required>
    <div class="valid-feedback">Looks good!</div>
  </div>
  <div class="col-md-4">
    <label for="vs2" class="form-label">Last name</label>
    <input type="text" class="form-control is-valid" id="vs2" required>
    <div class="valid-feedback">Looks good!</div>
  </div>
  <div class="col-md-4">
    <label for="vsu" class="form-label">Username</label>
    <div class="input-group has-validation">
      <span class="input-group-text">@</span>
      <input type="text" class="form-control is-invalid" id="vsu" required>
      <div class="invalid-feedback">Please choose a username.</div>
    </div>
  </div>
  <div class="col-md-6">
    <label for="vs3" class="form-label">City</label>
    <input type="text" class="form-control is-invalid" id="vs3" required>
    <div class="invalid-feedback">Please provide a valid city.</div>
  </div>
  <div class="col-md-3">
    <label for="vs4" class="form-label">State</label>
    <select class="form-select is-invalid" id="vs4" required>
      <option selected disabled value="">Choose...</option>
      <option value="1">one</option>
      <option value="2">two</option>
      <option value="3">three</option>
    </select>
    <div class="invalid-feedback">Please select a valid state.</div>
  </div>
  <div class="col-md-3">
    <label for="vs5" class="form-label">Zip</label>
    <input type="text" class="form-control is-invalid" id="vs5" required>
    <div class="invalid-feedback">Please provide a valid zip.</div>
  </div>
  <div class="col-12">
    <div class="form-check">
      <input class="form-check-input is-invalid" type="checkbox" value="" id="ic3" required>
      <label class="form-check-label" for="ic3">Agree to terms and conditions</label>
      <div class="invalid-feedback">You must agree before submitting.</div>
    </div>
  </div>
  <div class="col-12">
    <button class="btn btn-primary" type="submit">Submit form</button>
  </div>
</form>
```

Validation tooltip: - set tooltip for display validation messages, also for set tooltip make position relative of parent element of tooltip

Classes: - valid-tooltip, invalid-tooltip

Ex: -

```
<form class="row m-3 needs-validation" novalidate>
  <div class="col-md-4 position-relative">
    <label for="vt1" class="form-label">First name</label>
    <input type="text" class="form-control" id="vt1" required>
    <div class="valid-tooltip">Looks good!</div>
  </div>
  <div class="col-md-4 position-relative">
    <label for="vt2" class="form-label">Last name</label>
    <input type="text" class="form-control" id="vt2" required>
    <div class="valid-tooltip">Looks good!</div>
  </div>
  <div class="col-md-4 position-relative">
    <label for="vtu" class="form-label">Username</label>
    <div class="input-group has-validation">
      <span class="input-group-text">@</span>
      <input type="text" class="form-control" id="vtu" required>
      <div class="invalid-tooltip">Please choose a unique and valid username.</div>
    </div>
  </div>
  <div class="col-md-6 position-relative">
    <label for="vt3" class="form-label">City</label>
    <input type="text" class="form-control" id="vt3" required>
    <div class="invalid-tooltip">Please provide a valid city.</div>
  </div>
  <div class="col-md-3 position-relative">
    <label for="vt4" class="form-label">State</label>
    <select class="form-select" id="vt4" required>
      <option selected disabled value="">Choose...</option>
      <option value="1">one</option>
      <option value="2">two</option>
      <option value="3">three</option>
    </select>
    <div class="invalid-tooltip">Please select a valid state.</div>
  </div>
  <div class="col-md-3 position-relative">
    <label for="vt5" class="form-label">Zip</label>
    <input type="text" class="form-control" id="vt5" required>
    <div class="invalid-tooltip">Please provide a valid zip.</div>
  </div>
  <div class="col-12">
    <button class="btn btn-primary" type="submit">Submit form</button>
  </div>
</form>
```

Js: -

```
<script>
    (()=>{
    "use strict"
    const forms = document.querySelectorAll('.needs-validation');

    const f2=Array.from(forms)
    f2.forEach(validform);
    function validform(form)
    {
       form.addEventListener('submit',valid);
       function valid(event)
       {
          if(!form.checkValidity()){
          event.preventDefault();
          event.stopPropagation();
          }
       form.classList.add('was-validated');
       }
    }
    })()
</script>
```

Alerts: - provide contextual feedback message for typical user action with alert messages

Classes: - alert, alert-primary, alert-secondary, alert-success, alert-danger, alert-warning, alert-info, alert-light, alert-dark

Ex: -

```
<div class="alert alert-primary" role="alert">
   A simple primary alert—check it out!
</div>
```

Live example: - show alert when click on button

Class: - alert-dismissible

Ex: -

```
<div id="liveAlertPlaceholder"></div>
<button type="button" class="btn btn-primary" id="liveAlertBtn">Show live alert</button>
```

Js: -

```
const alertPlaceholder = document.getElementById('liveAlertPlaceholder')
const appendAlert = (message, type) => {
  const wrapper = document.createElement('div')
  wrapper.innerHTML = [
    `<div class="alert alert-${type} alert-dismissible" role="alert">`,
    `   <div>${message}</div>`,
    '   <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>',
    '</div>'
  ].join('')
  alertPlaceholder.append(wrapper)
}

const alertTrigger = document.getElementById('liveAlertBtn')
if (alertTrigger) {
  alertTrigger.addEventListener('click', () => {
    appendAlert('Nice, you triggered this alert message!', 'success')
  })
}
```

Alert link color: - it provide matching colored links within alert

Class: - alert-link

Ex: -

```
<div class="alert alert-success" role="alert">
  A simple primary alert—check it out! <a href="#" class="alert-link">click here</a>
</div>
```

Additional content: -

Class: - alert-heading

Ex: -

```
<div class="alert alert-success" role="alert">
  <h4 class="alert-heading">Well done!</h4>
  <p>
    Aww yeah, you successfully read this important alert message. This example text is going to run a
bit longer so that you can see how spacing within an alert works with this kind of content.
  </p>
  <hr>
  <p class="mb-0">
    Whenever you need to, be sure to use margin utilities to keep things nice and tidy.
  </p>
</div>
```

Alert icons: - use flexbox utilities and bootstrap icons to create alerts with icons, also copy and paste CDN link for bootstrap icons in page.

Ex: -

```
<div class="alert alert-primary d-flex align-items-center" role="alert">
  <i class="bi bi-check-circle-fill"></i>
  <div> An example alert with an icon</div>
</div>
```

Dismissing alert: - add .alert-dismissible to alert component and data-bs-dismiss="alert" attribute to close button.

Class: - alert-dismissible

Attribute: - data-bs-dismiss="alert"

Ex: -

```
<div class="alert alert-warning alert-dismissible fade show" role="alert">
  <strong>Holy guacamole!</strong> You should check in on some of those fields below.
  <button type="button" class="btn-close" data-bs-dismiss="alert"></button>
</div>
```

Scrollspy: - ignore non visible elements

Non-visible element: - target elements that are not visible (elements with .d-none, d-*-none) will be ignored and their nav items won't receive any .active class.

Scroll-spy instances initialized in a non-visible wrapper will ignore all target elements. Use the refresh method to check for observable elements once the wrapper becomes visible.

Set .d-*-block for element visible after provide breakpoint

Ex: -

```
<nav id="nbjs">
  <div class="nav nav-pills" id="nav-tab">
    <a class="nav-link active" data-bs-toggle="tab" href="#scjs1" data-bs-target="#con">
      Home
    </a>
    <a class="nav-link" data-bs-toggle="tab" href="#scjs2" data-bs-target="#con">
      Profile
    </a>
    <a class="nav-link" data-bs-toggle="tab" href="#scjs3" data-bs-target="#con">
      Contact
    </a>
    <a class="nav-link" data-bs-toggle="tab" href="#scjs4" data-bs-target="#con">
      About
    </a>
  </div>
```

```
</nav>
<div data-bs-spy="scroll" data-bs-smooth-scroll="true" class="bg-body-tertiary p-3 rounded-2"
tabindex="0" style="height: 200px; overflow-y: scroll;" id="con">
  <h4 id="scjs1">First heading</h4>
  <p> ... </p>
  <h4 id="scjs2" class="d-none d-md-block">Second heading</h4>
  <p> ... </p>
  <h4 id="scjs3">Third heading</h4>
  <p> ... </p>
  <h4 id="scjs4">Fourth heading</h4>
  <p> ... </p>
</div>
```

JS: -

```
document.querySelectorAll('#nav-tab>[data-bs-toggle="tab"]').forEach(el => {
  el.addEventListener('shown.bs.tab', () => {
    const target = el.getAttribute('data-bs-target')
    const scrollElem = document.querySelector(`${target}[data-bs-spy="scroll"]`)
    bootstrap.ScrollSpy.getOrCreateInstance(scrollElem).refresh()
  })
})
```

Explain: -
It first select every child of element with #nav-tab, that has [data-bs-toggle="tab"] attribute, to every that child, it will set event 'shown.bs.tab', when that event invoke, it will take value of attribute "data-bs-target", that will be id of targeted element, then `${target}[data-bs-spy="scroll"]` will select element will scrollspy and it will create or get instance of that element if that element is visible. Then refresh will called for creating that element

Popovers: - set popover to element in website

Enable popovers: - use javascript code for enable popovers, include popper.min.js to webpage before bootstrap.js or use bootstrap.bundle.min.js which contains popper.

JS: -

```
const popoverTriggerList = document.querySelectorAll('[data-bs-toggle="popover"]')
        const popoverList = [...popoverTriggerList].map(popoverTriggerEl => new bootstrap.Popover
(popoverTriggerEl))
```

Live demo: -

Setup: - set attribute data-bs-toggle="popover" to element, set data-bs-title for popover title and data-bs-content to set content that will display when popover appears.
Attribute: - data-bs-toggle="popover", data-bs-title, data-bs-content
Ex: -

```
<button type="button" class="btn btn-lg btn-danger" data-bs-toggle="popover" data-bs-
title="Popover" data-bs-content="And here's some amazing content. It's very engaging, Right?">Click
to toggle popover</button>
```

Directions: - set direction of popover with data-bs-placement

Attribute: - data-bs-placement="top", //left, right, top, bottom,
data-bs-container="body" //element that will contain popover

Ex: -

```
<button type="button" class="btn btn-secondary" data-bs-container="body" data-bs-toggle="popover"
data-bs-placement="top" data-bs-content="Top popover">
  Popover on top
</button>
```

Custom container: - specify custom container when parent element style interfere with popover

Setup: - set data-bs-container="button" in button container to see interfere with popover, in script set new popover with element that has set popover, and then set container value of that element popover with another element that will contain popover

Ex: -

```
<div class="border p-3 bg-primary" id="d1" style="width: 7rem;">
  <button type="button" class="btn btn-secondary" id="bpop" data-bs-container="button" data-bs-
toggle="popover" data-bs-content="here popover" data-bs-placement="top"> Popover </button>
</div>
```

JS: -

```
const popover = new bootstrap.Popover('#bpop', {
  container: 'body'
})
```

Custom popovers: - set custom popover style using CSS variables

Attribute: - data-bs-custom-class="custom-popover"

Ex: -

```
<style>
  .custom-popover{
    --bs-popover-max-width: 200px;
    --bs-popover-border-color: darkviolet;
    --bs-popover-header-bg: darkviolet;
    --bs-popover-header-color: white;
    --bs-popover-body-padding-x: 1rem;
    --bs-popover-body-padding-y: .5rem;
  }
</style>
```

```
<button type="button" class="btn btn-secondary" data-bs-toggle="popover" data-bs-placement="right" data-bs-custom-class="custom-popover" data-bs-title="Custom popover" data-bs-content="This popover is themed via CSS variables."> Custom popover </button>
```

Dismiss on next click: - use focus trigger to dissmiss popover when user click on other elements
Attribute: - data-bs-trigger="focus"
Ex: -

```
<a tabindex="0" class="btn btn-lg btn-danger" role="button" data-bs-toggle="popover" data-bs-trigger="focus" data-bs-title="Dismissible popover" data-bs-content="And here's some amazing content. It's very engaging, Right?">Dismissible popover</a>
```

Disabled elements: - disabled elements are not interactive, that popover cannot trigger when click or hover on it, use span as wrapper for trigger popover, set value of data-bs-trigger as "hover focus"
Attribute: - data-bs-trigger="hover focus"
Ex: -

```
<span class="d-inline-block" tabindex="0" data-bs-toggle="popover" data-bs-trigger="hover focus" data-bs-content="Disabled popover">
   <button class="btn btn-primary" type="button" disabled>Disabled button</button>
</span>
```

Toasts: - pust notification using toast, lightweight alert messages

Setup: - set button for show toast and set id to that button, for toast set wrapper element .toast-container, in that container set .toast container with id, in that container set .toast-header for header information, also in .toast-header set button with .btn-close class and data-bs-dismiss="toast" attribute. After header set container .toast-body for display content in toast.
Classes: - toast-container, toast, toast-header, toast-body
Attribute: - data-bs-dismiss="toast"
Ex: -

```
<button type="button" class="btn btn-primary" id="liveToastBtn">Show live toast</button>

<div class="toast-container position-fixed bottom-0 end-0 p-3">
   <div id="liveToast" class="toast">
      <div class="toast-header">
         <img src="../3css/res/background.jpg" class="rounded me-2" style="width:30px;">
         <strong class="me-auto">Bootstrap</strong>
         <small> 11 mins ago</small>
         <button type="button" class="btn-close" data-bs-dismiss="toast"></button>
      </div>
      <div class="toast-body">
         Hello, world! This is a toast message.
      </div>
   </div>
</div>
```

JS: -

```
const toastTrigger = document.getElementById('liveToastBtn')
const toastLive = document.getElementById('liveToast')
if (toastTrigger) {
  const toastBootstrap = bootstrap.Toast.getOrCreateInstance(toastLive)
  toastTrigger.addEventListener('click', () => {
    toastBootstrap.show()
  })
}
```

Explain: - target button for trigger toast and .toast container, if toast button exsist, it will create or get instance of toast by passed toaste element, then it will set click event for toast button, that will called show method when button get clicked

Translucent: - toasts are slightly translucent to blend in what is at their background

Ex: -

```
<body class="bg-dark">
  <button type="button" class="btn btn-primary" id="liveToastBtn">Show live toast</button>

  <div class="toast-container position-fixed bottom-0 end-0 p-3"> ... </div>
</body>
```

Stacking: - stack toasts by wrapping them in .toast-container, which will vertically add some spacing.

Note: - initialise all toast with javascript for display them.

Ex: -

```
<div class="toast-container position-fixed bottom-0 end-0 p-3">
  <div id="liveToast" class="toast"> ... </div>
  <div id="liveToast2" class="toast"> ... </div>
</div>
```

Custom content: - customize toast by removing sub-components or adding aditional content

Ex: -

```
<div class="toast-container position-fixed bottom-0 end-0 p-3">
  <div id="liveToast" class="toast align-items-center">
    <div class="d-flex">
      <div class="toast-body"> Hello, world! This is a toast message. </div>
      <button type="button" class="btn-close me-2 m-auto" data-bs-dismiss="toast"></button>
    </div>
  </div>
  <div id="liveToast2" class="toast">
    <div class="toast-body">
      Hello, world! This is a toast message.
      <div class="mt-2 pt-2 border-top">
```

```
            <button type="button" class="btn btn-primary btn-sm"> Take action </button>
            <button type="button" class="btn btn-secondary btn-sm" data-bs-dismiss="toast">
               Close
            </button>
         </div>
      </div>
   </div>
</div>
```

Color schemes: - use color and background-color utilities for change color of toast and also change color close button using .btn-close-white utility.

Ex: -

```
<div id="liveToast" class="toast align-items-center text-bg-primary">
   <div class="d-flex">
      <div class="toast-body"> Hello, world! This is a toast message. </div>
      <button type="button" class="btn-close btn-close-white me-2 m-auto" data-bs-dismiss="toast">
</button>
   </div>
</div>
```

Placement: - use position and placing utilities, flex utilites can also used

Values: - top-0 start-0, top-0 start-50 translate-middle-x, top-0 end-0, top-50 start-0 translate-middle-y, top-50 start-50 translate-middle, top-50 end-0 translate-middle-y, bottom-0 start-0, bottom-0 start-50 translate-middle-x, bottom-0 end-0

Ex: -

```
<div class="toast-container position-fixed top-0 start-0 p-3"> ... </div>
```

Ex: - use flex utilities

```
<div class="toast-container position-fixed top-50 start-50 translate-middle d-flex justify-content-center
align-items-center w-100 p-3"> ... </div>
```

Accessibility: - set display duration or auto hide with corresponding attributes.

Attribute: - data-bs-delay="10000" //time in milliseconds, data-bs-autohide="false"

Ex: - set delay time for hide toast

```
<div id="liveToast" class="toast text-bg-primary" data-bs-delay="10000">
   ... ... ...
</div>
```

Ex: - set data-bs-autohide to false for close manually also need implement close button for close

```
<div id="liveToast" class="toast text-bg-primary" data-bs-autohide="false">
   ... ... ...
   <button type="button" class="btn-close btn-close-white me-2 m-auto" data-bs-dismiss="toast">
</button>
```

```
</div>
```

Modal: - it is javascript plugin to add dialogs to site for lightboxes, user notifications

Live demo: -

Setup: - set button with attribute data-bs-toggle="modal" and target model id, using data-bs-target. After button set container .modal with id that button is targetting, also set tabindex="-1" for not focusing modal, in that container set .modal-dialog container, in that set .model-content for set content of modal, in that container set .modal-header for heading, in that container set .modal-title for set title, after it set close button with data-bs-dismiss="modal" attribute, after header container set .modal-body container for display content, after it set .modal-footer container for footer content.

Classes: - modal, modal-dialog, modal-content, modal-header, modal-title, modal-body, modal-footer

Attributes: - data-bs-toggle="modal", data-bs-target, tabindex="-1", data-bs-dismiss="modal"

Ex: -

```
<button type="button" class="btn btn-primary" data-bs-toggle="modal" data-bs-target="#exMod">
  Launch demo modal
</button>

<div class="modal fade" id="exMod" tabindex="-1">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h1 class="modal-title fs-5">Modal title</h1>
        <button type="button" class="btn-close" data-bs-dismiss="modal" id="myInput"></button>
      </div>
      <div class="modal-body"> ... </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
        <button type="button" class="btn btn-primary">Save changes</button>
      </div>
    </div>
  </div>
</div>
```

JS: - due to how HTML5 defines semantics autofocus attribute has no effect in bootstrap modals, use custom javascript for achieve autofocus effect.

```
const myModal = document.getElementById('exMod')
const myInput = document.getElementById('myInput')

myModal.addEventListener('shown.bs.modal', () => {
  myInput.focus()
})
```

Explain: - target model component and input or button that will get focus using javascript set 'shown.bs.modal' event using addEventListener to that modal, in function parameter of addEventListener create or pass function that call focus() function for input or button

Static backdrop: - set modal backdrop as static, for modal not close when click outside of it.
Attribute: - data-bs-backdrop="static"
Ex: -

```
<div class="modal fade" id="exMod" data-bs-backdrop="static" tabindex="-1"> ... </div>
```

Scrolling long content: - when modal content become large that viewport or device, it scroll independant of the page itself. Also create scrollable modal that allows scrolling modal body using .modal-dialog-scrollable to .modal-dialog.
Class: - modal-dialog-scrollable
Ex: -

```
<div class="modal fade" id="exMod" tabindex="-1">
   ...
   <div class="modal-body" style="height: 700px;"> ... </div>
   ...
</div>
```

Ex: - make modal body scrollable

```
<div class="modal fade" id="exMod" tabindex="-1">
   <div class="modal-dialog modal-dialog-scrollable"> ... </div>
</div>
```

Vertically centered modal: - add .modal-dialog-centered to .modal-dialog for vertically centered modal.
Classes: -
Ex: -

```
<div class="modal fade" id="exMod" tabindex="-1">
   <div class="modal-dialog modal-dialog-scrollable modal-dialog-centered"> ... </div>
</div>
```

Tooltips and popovers in modal: -
Ex: -

```
<div class="modal fade" id="exMod" tabindex="-1">
   ...
   <div class="modal-body">
      <h2 class="fs-5">Popover in a modal</h2>
      <p>this <button class="btn btn-secondary" data-bs-toggle="popover" title="Popover title" data-bs-content="Popover body content is set in this attribute.">button</button> triggers a popover on click </p>
      <hr>
```

```
        <h2 class="fs-5">Tooltips in a modal</h2>
        <p><a href="#" data-bs-toggle="tooltip" title="Tooltip">This link</a> and <a href="#" data-bs-toggle="tooltip" title="Tooltip">that link</a> have tooltips on hover </p>
    </div>
</div>
    ...
</div>
```

Using grid in modal: - use .container-fluid in .modal-body for normal grid system.

Ex: -

```
<div class="modal fade" id="exMod" tabindex="-1">
    ...
    <div class="modal-body">
        <div class="container-fluid">
            <div class="row">
                <div class="col-md-4">.col-md-4</div>
                <div class="col-md-4 ms-auto">.col-md-4 .ms-auto</div>
            </div>
            <div class="row">
                <div class="col-md-3 ms-auto">.col-md-3 .ms-auto</div>
                <div class="col-md-2 ms-auto">.col-md-2 .ms-auto</div>
            </div>
            <div class="row">
                <div class="col-md-6 ms-auto">.col-md-6 .ms-auto</div>
            </div>
            <div class="row">
                <div class="col-sm-9">
                    Level 1: .col-sm-9
                    <div class="row">
                        <div class="col-8 col-sm-6"> Level 2: .col-8 .col-sm-6 </div>
                        <div class="col-4 col-sm-6"> Level 2: .col-4 .col-sm-6 </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
    ...
</div>
```

Varying modal content: - set multiple buttons for trigger same modal with slightly diferrent content, use event.relatedTarget and data-bs-whatever attribute to vary the contents of modal depend on which button is clicked.

Attribute: - data-bs-whatever

Setup: - set three buttons with data-bs-whatever with different data, and target all button to single modal and set attribute data-bs-toggle="modal" to toggle modal, and create modal with .modal-title and in .modal-body set input element

Ex: -

```
<button type="button" class="btn btn-primary" data-bs-toggle="modal" data-bs-target="#exMod2"
data-bs-whatever="@mdo">Open modal for @mdo</button>
<button type="button" class="btn btn-primary" data-bs-toggle="modal" data-bs-target="#exMod2"
data-bs-whatever="@fat">Open modal for @fat</button>
<button type="button" class="btn btn-primary" data-bs-toggle="modal" data-bs-target="#exMod2"
data-bs-whatever="@getbootstrap">Open modal for @getbootstrap</button>

<div class="modal fade" id="exMod2" tabindex="-1">
  <div class="modal-dialog modal-dialog-scrollable">
    <div class="modal-content">
      <div class="modal-header">
        <h1 class="modal-title fs-5">New message</h1>
        <button type="button" class="btn-close" data-bs-dismiss="modal"></button>
      </div>
      <div class="modal-body">
        <form>
          <div class="mb-3">
            <label for="recipient-name" class="col-form-label">Recipient:</label>
            <input type="text" class="form-control" id="recipient-name">
          </div>
          <div class="mb-3">
            <label for="message-text" class="col-form-label">Message:</label>
            <textarea class="form-control" id="message-text"></textarea>
          </div>
        </form>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
        <button type="button" class="btn btn-primary">Send message</button>
      </div>
    </div>
  </div>
</div>
```

Js: -

```
const exampleModal = document.getElementById('exMod2')

if (exampleModal) {
  exampleModal.addEventListener('show.bs.modal', event => {
```

```
  // Button that triggered the modal
  const button = event.relatedTarget

  // Extract info from data-bs-* attributes
  const recipient = button.getAttribute('data-bs-whatever')

  // Update the modal's content.
  const modalTitle = exampleModal.querySelector('.modal-title')

  const modalBodyInput = exampleModal.querySelector('.modal-body input')

  modalTitle.textContent = `New message to ${recipient}`

  modalBodyInput.value = recipient

  })
}
```

Explain: - first target element in javascript using document.getElementById method, then check if modal component is exist using if statement, then set 'show.bs.modal' event to that modal with addEventListener method, in function that method use event.relatedTarget to target which button is clicked, and store that button element in contant, from that button take value of 'data-bs-whatever' attribute using getAttribute method and stored that value in another constant, then target '.modal-title' and '.modal-body input' using querySelector methods, set string value to '.modal-title' element using .textContent property that take from 'data-bs-whatever', also set same value to targeted input using value property.

Toggle between modals: - toggle between two modals.
Note: - mutiple modals cannot open at same time
Setup: - set button for open and target first modal, in first modal setup button for target and open second modal, and in second modal setup button to target and open first modal also add close buttons to both modal for close.
Ex: -

```
<button class="btn btn-primary" data-bs-target="#exMod3" data-bs-toggle="modal">
  Open first modal
</button>

<div class="modal fade" id="exMod3" tabindex="-1">
  <div class="modal-dialog modal-dialog-centered">
    <div class="modal-content">
      <div class="modal-header">
        <h1 class="modal-title fs-5">Modal 1</h1>
        <button type="button" class="btn-close" data-bs-dismiss="modal"></button>
      </div>
      <div class="modal-body">
```

```
                Show a second modal and hide this one with the button below.
            </div>
            <div class="modal-footer">
                <button class="btn btn-primary" data-bs-target="#exMod4" data-bs-toggle="modal">
                    Open second modal
                </button>
            </div>
        </div>
    </div>
</div>
<div class="modal fade" id="exMod4" tabindex="-1">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content">
            <div class="modal-header">
                <h1 class="modal-title fs-5">Modal 2</h1>
                <button type="button" class="btn-close" data-bs-dismiss="modal"></button>
            </div>
            <div class="modal-body" style="height: 800px">
                Hide this modal and show the first with the button below.
            </div>
            <div class="modal-footer">
                <button class="btn btn-primary" data-bs-target="#exMod3" data-bs-toggle="modal">
                    Back to first
                </button>
            </div>
        </div>
    </div>
</div>
```

Remove animation: - remove .fade class from modal to remove animation.

Ex: -

```
<div class="modal" id="exMod3" tabindex="-1"> ... </div>
```

Optional sizes: - modals have three optional sizes

Classes: - modal-xl, modal-lg, modal-sm

Ex: -

```
<div class="modal fade modal-sm" id="exMod4" tabindex="-1"> ... </div>
```

Fullscreen modal: - set fullscreen modal when viewport is before breakpoint, .modal-fullscreen is set alway fullscreen.

Classes: - modal-fullscreen, modal-fullscreen-sm-down, modal-fullscreen-md-down, modal-fullscreen-lg-down, modal-fullscreen-xl-down, modal-fullscreen-xxl-down

Ex: -

```
<div class="modal fade" id="exMod3" tabindex="-1">
  <div class="modal-dialog modal-fullscreen-md-down"> ... </div>
</div>
```