



INTRODUCTION TO BOOTSTRAP 5: UTILITIES & CORE COMPONENTS

ABSTRACT

This lecture introduces Bootstrap setup, containers, utility classes, image styling, icons, and tables. It further covers the grid system, alerts, buttons, spinners, dropdowns, cards, collapse, and accordion, enabling quick, responsive, and professional UI development.

Bootstrap 5.0

Bootstrap is a free and open-source tool collection for creating responsive websites and web applications. It is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites. Nowadays, the websites are perfect for all browsers (IE, Firefox, and Chrome) and for all sizes of screens (Desktop, Tablets, Phablets, and Phones).

All thanks to Bootstrap developers – **Mark Otto** and **Jacob Thornton** of Twitter, though it was later released as an open-source framework in August 2011 on GitHub..

Importance of Bootstrap

By using this framework, we can easily manipulate the styling of any web page, like font style, text color, background color, flex, grid system, etc. Bootstrap Version 4 & Version 5 are the most popular versions. There are lots of other CSS frameworks like Tailwind CSS, Bulma, and Foundation but among them, this framework is the most popular because of below mentioned features:

- **Easy to use:** Anyone with basic HTML & CSS knowledge can use it.
- **Responsive:** Automatically adjusts to all screen sizes.
- **Mobile-first:** Bootstrap 3 and later versions use mobile-first design by default.
- It is a free and open-source framework available on www.getbootstrap.com

Responsive Website

A website is called **responsive** when it can **automatically adjust its layout and content** to look good on all devices:

- Smartphones
- Tablets
- Laptops
- Desktops

Responsive design adapts to different screen sizes using **flexible grids, media queries, and fluid images**.



Methods to Link Bootstrap

Bootstrap can link in two ways:

1. Use Latest Bootstrap via CDN from www.getbootstrap.com
2. Download Bootstrap

1. Use Bootstrap via CDN

- No download required.
- Faster loading due to distributed servers.
- Benefits:
 - File may already be cached in the user's browser.
 - CDN delivers file from the nearest server, faster loading.

What is a CDN?

A **Content Delivery Network** is a system of distributed servers that quickly delivers: Webpages, Images, CSS files, JavaScript files.

Example:

- A web server in **Delhi** hosts your content.
- A user in **Ahmedabad** requests it.
- CDN serves the file from the nearest server → faster response.

What is jsDelivr?

jsDelivr is a free Content Delivery Network (CDN) used to load popular libraries like Bootstrap, jQuery, Font Awesome, etc.

It helps websites load files **faster** by delivering them from the server closest to the user.
jsDelivr is a fast, free CDN that provides ready-to-use links for CSS and JS libraries (like **Bootstrap**).

Links and scripts for bootstrap

Type	Code (write these highlighted link and script in exam)	Description	When to Use
CSS (Required)	<code><link rel="stylesheet" href="bootstrap.min.css"></code>	Loads Bootstrap's styles (layout, components, utilities). Must be included in every Bootstrap project.	Always use. Needed for all Bootstrap designs and components.
Bundle JS (Recommended)	<code><script src="bootstrap.bundle.min.js"> </script></code>	Includes Bootstrap JS + Popper in one file. Simple and efficient.	Use this most of the time. Best for beginners and general projects.
Separate JS	<code><script src="popper.min.js"></script> <script src="bootstrap.min.js"></script></code>	Popper and Bootstrap JS are loaded separately. Allows manual control of dependencies.	Use only when you already include Popper separately or need advanced customization.

Code(use these to integrate Bootstrap in your code)

CSS (Required)

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"  
integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwpSpd3yD65VohhpwCOmLASjC"  
crossorigin="anonymous">
```

Introduction to Bootstrap 5: Utilities & Core Components

Bundle JS (Recommended)

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcW6ZMFYlzcLA8NI+NtUVF0sA7MsXsP1UyJoMp4YLEuNsFAP+JcXn/tWtIaxVXM" crossorigin="anonymous"></script>
```

Separate JS

```
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.2/dist/umd/popper.min.js" integrity="sha384-IQsoLXI5PILFhosVNubq5LC7Qb9DXgDA9i+tQ8Zj3iwWAwPtgFTxbJ8NT4GN1R8p" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.min.js" integrity="sha384-cVKIPhGWiC2Al4u+LWgxfKTRIcfu0JTxR+EQDz/bgldoEyl4H0zUF0QKbrJ0EcQF" crossorigin="anonymous"></script>
```

What is integrity?

- It is used for Subresource Integrity (SRI).
- It ensures the file (CSS or JS) you are loading from a CDN has not been changed, hacked, or corrupted.
- integrity = security check to ensure the file is safe and unchanged.

What is crossorigin?

- It is related to CORS (Cross-Origin Resource Sharing).
- It tells the browser how to handle requests for files loaded from other domains (like a CDN).
- When using integrity, browsers often need crossorigin="anonymous" so they can properly verify the file.

Example:

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Bootstrap demo</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.8/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-sRI4kxFvY47J16cr9ZwB07vP4J8+LH7qKQnuqkulAvNWLzeN8tE5YBujZqJLB" crossorigin="anonymous">
  </head>
  <body>
    <h1>Hello, world!</h1>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.8/dist/js/bootstrap.bundle.min.js" integrity="sha384-FKyoEForCGlyvwx9Hj09JcYn3nv7wiPVlz7YYwJrWVcXK/BmnVDxM+D2scQbITxi" crossorigin="anonymous"></script>
  </body>
</html>
```

Location of <link>: Always in the <head> section.

- Because CSS must load **before** the page is displayed to avoid unstyled or broken layouts.

In short: <link> goes in <head> so the page loads with proper styling.

Location of <script>

- **In <body> (Recommended):** Loads after the content → page loads faster.
- **In <head>:** Loads before content → use only if a script must run early.

In short: put scripts at the end of <body> for better performance.

2. Download Bootstrap

- From: getbootstrap.com
- You can host the files locally.

How to Download Bootstrap

1. Go to getbootstrap.com
2. Click **Download**
3. Select **Compiled CSS and JS**
4. Extract the downloaded ZIP file
5. Inside the folder, find bootstrap.min.css and bootstrap.min.js
6. Copy these files into your project (e.g., css/ and js/ folders)
7. Link them in your HTML:

```
<link rel="stylesheet" href="css/bootstrap.min.css">
<script src="js/bootstrap.bundle.min.js"></script>
```

When to Use Downloaded Bootstrap?

- When working **offline**
- For **college assignments** or practical exams
- When you want to **customize** Bootstrap files
- When the project runs on an **internal network**
- When you don't want to rely on a CDN

When to Use CDN Instead?

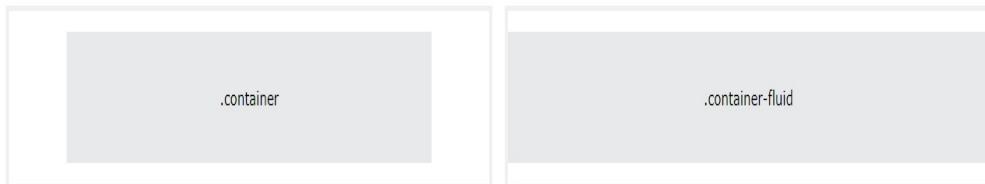
- When you need **faster loading**
- For **quick setup**
- When internet is available
- For better **performance and caching**

Bootstrap 5 Container

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

1. The **.container** (Fixed-width) class provides a responsive fixed width container. It gives your page a clean, readable layout and is used as the base for the grid system.
2. The **.container-fluid** (Full-width) class provides a full width container, spanning the entire width of the viewport. It is used when you want edge-to-edge designs, full-width sections, or layouts that fill all horizontal space.

Class	Width
.container	Has a fixed width and stays centered on the page.
.container-fluid	Always 100% full width , stretching across the entire screen.



Screen Size	Class	Breakpoint (viewport width)	.container Max-Width	.container-{breakpoint} Behavior
Extra small	container-xs	< 576px	100% (fluid)	.container. 100% width <576px
Small	container-sm	≥ 576px	540px	.container-sm becomes fixed-width at sm and larger; fluid below sm.
Medium	container-md	≥ 768px	720px	.container-md is fixed-width at md and larger; fluid below md.
Large	container-lg	≥ 992px	960px	.container-lg is fixed-width at lg and larger; fluid below lg.
Extra large	container-xl	≥ 1200px	1140px	.container-xl is fixed-width at xl and larger; fluid below xl.
Extra extra large	container-xxl	≥ 1400px	1320px	.container-xxl is fixed-width at xxl and larger; fluid below xxl.

Example

```
<body>
  <div class="container">
    <h1>My fixed container Bootstrap Page</h1>
    <p>Chapter-6 Introduction to Bootstrap</p>
  </div>
  <div class="container-fluid">
    <h1>My Fluid container Bootstrap Page</h1>
    <p>This part is inside a .container-fluid class.</p>
  </div>
</body>
```

Output:

My fixed container Bootstrap Page

Chapter-6 Introduction to Bootstrap

My Fluid container Bootstrap Page

This part is inside a .container-fluid class.

Note:

1. .container

- Responsive **fixed-width** container.
- Width changes at each breakpoint.

2. .container-fluid

- Always **100% width** (no breakpoints).

3. .container-{breakpoint}

These become **fixed-width only at that breakpoint and above**.
Below that breakpoint, they behave like .container-fluid.

Example:

- .container-md is **fluid** for xs and sm
- Becomes **720px fixed** at md ($\geq 768\text{px}$) and up.

Bootstrap Utility & Sizing Classes

Spacing Utilities : margin & padding

Token	Meaning
mX1-X2	margin
pX1-X2	padding

X1 = Side (where it applies)

X2= Size (spacing => (1 rem = 16 px)]

Token	Applies To
(blank)	all sides
t	top
b	bottom
s	start (left in LTR, right in RTL)
e	end (right in LTR, left in RTL)
x	left + right
y	top + bottom

Token	Value (default)
0	0
1	0.25rem
2	0.5rem
3	1rem
4	1.5rem
5	3rem
auto	auto (margin only)

Example: {property}{side}-{size}

- m-3 → margin 1rem on all sides
- mt-2 → margin-top 0.5rem
- px-4 → padding left+right 1.5rem
- ms-0 → remove start margin
- mx-auto → horizontal auto margin

Width Classes (w-*)

Class	What It Means	Description
w-25	width: 25%	Element takes 25% of its parent's width.
w-50	width: 50%	Element takes half of the parent's width.
w-75	width: 75%	Element takes 75% of the parent's width.
w-100	width: 100%	Element stretches to the full width of its parent.
w-auto	width: auto	Width adjusts to the content's natural size.

Height Classes (h-*)

Class	What It Means	Description
h-25	height: 25%	Element's height becomes 25% of its parent container.
h-50	height: 50%	Half the height of the parent.
h-75	height: 75%	75% of the parent's height.
h-100	height: 100%	Fills the entire height of the parent.
h-auto	height: auto	Height adapts based on content.

Bootstrap Text Colors

Class	Description (Expanded Meaning)
text-primary	Applies the theme's main "primary" color (usually blue).
text-secondary	Applies a secondary accent color (usually gray).
text-success	Applies the success color (green), often used for positive messages.
text-danger	Applies the danger color (red), used for errors or warnings.
text-warning	Applies the warning color (yellow/orange), used for caution messages.
text-info	Applies the info color (teal/light blue), used for informational notes.
text-dark	Applies a dark gray/black text color.
text-light	Applies a very light/white-ish text color.
text-muted	Applies a soft, faded gray color used for less important text.
text-white	Applies pure white text color (#fff) to the element. Useful for dark backgrounds, headers, footers, banners, buttons, and overlays.

Example:

```
<div class="container">
<h2>Contextual Colors</h2>
<p>Use the contextual classes to provide "meaning through colors":</p>
<p class="text-muted">This text is muted.</p>
<p class="text-primary">This text is important.</p>
<p class="text-success">This text indicates success.</p>
<p class="text-info">This text represents some information.</p>
<p class="text-warning">This text represents a warning.</p>
<p class="text-danger">This text represents danger.</p>
<p class="text-secondary">Secondary text.</p>
<p class="text-dark">This text is dark grey.</p>
<p class="text-light">This text is light grey (on white background).</p>
<p class="text-white">This text is white (on white background).</p>
</div>
```

Contextual Colors

Use the contextual classes to provide "meaning through colors":

This text is muted.

This text is important.

This text indicates success.

This text represents some information.

This text represents a warning.

This text represents danger.

Secondary text.

This text is dark grey.

This text is light grey (on white background).

NOTE (*Ref) : The **!important** rule in CSS is used to add more importance to a property/value than normal.

In fact, if you use the !important rule, it will override ALL previous styling rules for that specific property on that element.

We can also add 50% opacity white and black text colors using “text-white-50” (background must be dark to see changes) and “text-black-50” classes (background must be light to see changes).

50% opacity:

```
<div class="container">
  <h2>Opacity Text Colors</h2>
  <p class="text-success" style="color:black !important">This text indicates success.</p>
  <p>Add 50% opacity for black or white text with the .text-black-50 or .text-white-50
  classes:</p>
  <p class="text-black-50">Black text with 50% opacity on white background</p>
  <p class="text-white-50 bg-dark">White text with 50% opacity on black background</p>
</div>
```

This text indicates success.

This text is black 50% .

This text is white 50%

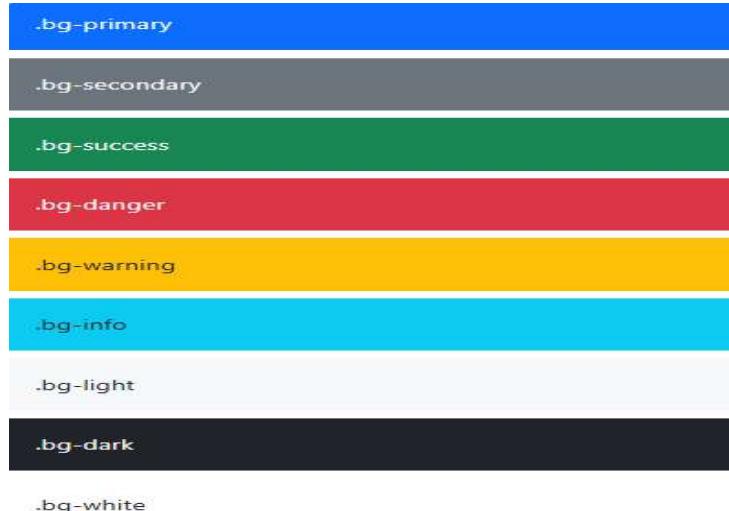
Background Colors

Class	Text Color Used	Description
bg-primary	text-white	Applies the primary theme background color (usually blue).
bg-secondary	text-white	Applies the secondary gray background color.
bg-success	text-white	Applies the success (green) background color.
bg-danger	text-white	Applies the danger (red) background color.
bg-warning	text-dark	Applies the warning (yellow/orange) background color.
bg-info	text-dark	Applies the info (teal/light-blue) background color.
bg-light	text-dark	Applies a very light gray/white background.
bg-dark	text-white	Applies a dark gray/black background.
bg-white	text-dark	Sets the background to pure white (#fff).
bg-transparent	text-dark	Makes the background fully transparent (no color).

Introduction to Bootstrap 5: Utilities & Core Components

Example:

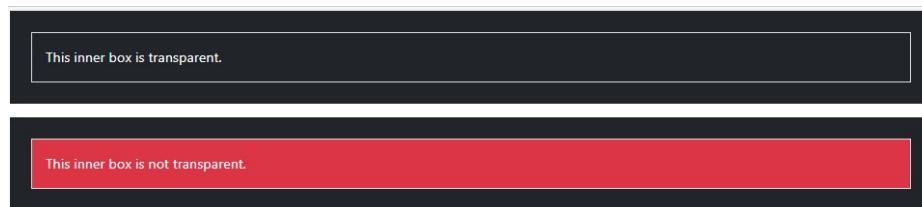
```
<div class="p-3 mb-2 bg-primary text-white">.bg-primary</div>
<div class="p-3 mb-2 bg-secondary text-white">.bg-secondary</div>
<div class="p-3 mb-2 bg-success text-white">.bg-success</div>
<div class="p-3 mb-2 bg-danger text-white">.bg-danger</div>
<div class="p-3 mb-2 bg-warning text-dark">.bg-warning</div>
<div class="p-3 mb-2 bg-info text-dark">.bg-info</div>
<div class="p-3 mb-2 bg-light text-dark">.bg-light</div>
<div class="p-3 mb-2 bg-dark text-white">.bg-dark</div>
<div class="p-3 mb-2 bg-white text-dark">.bg-white</div>
```



Example of bg-transparent

```
<div class="container bg-dark w-50 p-4 text-white mb-3">
  <div class="bg-transparent p-3 border">
    This inner box is transparent.
  </div>
</div>

<div class="container bg-dark w-50 p-4 text-white">
  <div class="bg-danger p-3 border">
    This inner box is not transparent.
  </div>
</div>
```



Introduction to Bootstrap 5: Utilities & Core Components

Typography and text utilities

Class	Category	Description
text-lowercase	Text Transform	Converts all letters in the text to lowercase.
text-uppercase	Text Transform	Converts all letters in the text to uppercase.
text-capitalize	Text Transform	Capitalizes the first letter of each word , leaving other letters unchanged.
text-decoration-none	Text Decoration	Removes underline and other text decoration (useful for links).
text-decoration-underline	Text Decoration	Adds an underline under the text.
text-decoration-line-through	Text Decoration	Adds a line through the text (strikethrough).
text-start	Alignment	Aligns text to the left (start). RTL-aware.
text-center	Alignment	Centers the text horizontally.
text-end	Alignment	Aligns text to the right (end). RTL-aware.
fs-1 to fs-6	Font Size	Sets the font size. fs-1 is largest, fs-6 is smallest. Matches HTML heading scale.
fw-bold	Font Weight	Makes the text bold.
fw-bolder	Font Weight	Makes the text bolder than the parent element.
fw-light	Font Weight	Applies a light font weight.
fw-lighter	Font Weight	Makes the text lighter than the parent element.
fst-italic	Font Style	Applies italic styling.
fst-normal	Font Style	Removes italic styling; returns to normal.

Shadows

Class	Detailed Description
shadow-sm	Adds a small, subtle shadow , giving a light lifted look without strong depth.
shadow	Applies a standard (medium) shadow , commonly used for cards and floating elements.
shadow-lg	Adds a large, deeper shadow to create strong elevation and highlight prominence.

Borders

Class	Detailed Description
border	Adds a 1px solid border to all four sides of the element using the default border color.
border-0	Removes all borders , overriding any border styles the element may have.
border-top	Adds a border only to the top edge of the element.
border-bottom	Adds a border only to the bottom edge of the element.
border-start	Adds a border to the start side (left in LTR, right in RTL).
border-end	Adds a border to the end side (right in LTR, left in RTL).
border-primary	Applies the primary theme color to the border.
border-secondary	Applies the secondary theme color to the border.
border-success	Applies the success (green) color to the border.
border-danger	Applies the danger (red) color to the border.
border-warning	Applies the warning (yellow/orange) color to the border.
border-info	Applies the info (teal/light-blue) color to the border.
border-dark	Applies a dark-colored border.
border-light	Applies a very light-colored border.
border-white	Applies a white border (useful on dark backgrounds).
border-1	Sets border thickness to thin (1px) .
border-2	Sets border thickness to medium-thin .
border-3	Medium thickness, more visible border.
border-4	Thick border for emphasis.
border-5	Very thick, bold border (strong visual emphasis).
rounded	Adds standard rounded corners (0.25rem).
rounded-circle	Makes the element fully circular using border-radius: 50% (works best when width = height).

Example

```
<style>
.box {display:inline-block;height:50px;width:50px}
</style>
<div>
    <!-- border color + width -->
    <span class="box border border-primary border-1"></span>
    <span class="box border border-secondary border-2"></span>
    <span class="box border border-success border-3"></span>

    <!-- border color + width + radius -->
    <span class="box border border-danger border-4 rounded"></span>

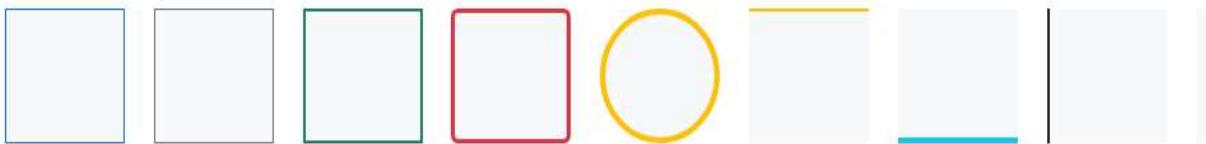
    <!-- border color + width + circle -->
    <span class="box border border-warning border-5 rounded-circle"></span>

    <!-- border color + top-only + width -->
    <span class="box border-top border-warning border-3"></span>

    <!-- border color + bottom-only + width -->
    <span class="box border-bottom border-info border-5"></span>

    <!-- border color + start-only + width -->
    <span class="box border-start border-dark border-3"></span>

    <!-- border color + end-only + width-->
    <span class="box border-end border-danger border-3"></span>
</div>
```



Task: Create First Web Page With Bootstrap

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">

<!-- Bootstrap 5 CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<title>All Utility Classes Example</title>
</head>
<body>

<!-- Full width section -->
<div class="container-fluid bg-dark py-4">
  <h2 class="text-center text-white text-uppercase">bootstrap utilities demo</h2>
</div>

<!-- Main section using all classes -->
<div class="container my-5">
  <div class="p-4 bg-warning text-dark border border-danger border-5 rounded shadow-lg text-center text-capitalize mx-auto w-50">
    <h3 class="mb-3">all utilities in one box</h3>
    <p class="mb-1">
      Lorem ipsum dolor sit amet consectetur adipisicing elit. Accusamus labore quas, beatae delectus nisi dolorem quod neque iste ea praesentium, cum, eligendi amet provident culpa ipsam tenetur repudiandae? Quod, itaque.
    </p>
  </div>
</div>

<!-- Bootstrap JS -->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

BOOTSTRAP UTILITIES DEMO

All Utilities In One Box

Display

Class	Description (Expanded & Easy Explanation)
d-none	Completely hides the element. It does not take up any space on the page.
d-block	Makes the element a block-level element . It starts on a new line and takes the full width available.
d-inline	Makes the element an inline element . It does not start on a new line and only takes the width of its content.
d-inline-block	Acts like inline (no line break) but allows setting width and height like a block element.
d-flex	Turns the element into a flex container , enabling Flexbox layout for arranging child elements.
d-inline-flex	Same as d-flex, but the element behaves inline (no line break before/after).
d-grid	Turns the element into a CSS Grid container , enabling grid layout for its children.

Positioning

Class	Detailed Description
position-static	Default position. The element stays in normal document flow and is not affected by top/start/end/bottom offsets.
position-relative	The element stays in the normal flow but can be shifted using top/start/end/bottom without affecting other elements' layout. Often used as a containing parent for absolutely positioned children.
position-absolute	The element is removed from normal flow and positioned relative to its nearest positioned ancestor (not static). Uses offsets to place itself precisely.
position-fixed	Stays fixed relative to the viewport . Does not move even when the page scrolls. Common for sticky headers, floating buttons, and alerts.
position-sticky	Acts like relative until a scroll threshold is reached, then becomes fixed at the top (or side) based on offsets. Useful for sticky headers or sidebars.
top-0 / top-50 / top-100	Moves the element down from the top edge of its positioned container by 0%, 50%, or 100%.
bottom-0 / bottom-50 / bottom-100	Moves the element upward from the bottom by the given percentage.
start-0 / start-50 / start-100	Offsets from the start side (left in LTR, right in RTL).
end-0 / end-50 / end-100	Offsets from the end side (right in LTR, left in RTL).

Introduction to Bootstrap 5: Utilities & Core Components

Images

- ✓ **Bootstrap** offers different classes for images to make their appearance better and also to make them responsive.
- ✓ Making an image responsive means it should scale according to its parent element.
- ✓ That is, the size of the image should not overflow its parent element and will grow and shrink according to the change in the size of its parent without losing its aspect ratio.
- ✓ The different classes available in Bootstrap for images are as explained below:

Class	Description	Effect on Image
img-fluid	Makes image responsive.	Image adjusts size automatically
rounded	Gives the image small rounded corners.	Slightly curved edges
rounded-circle	Makes the image a perfect circle	Circular profile photo
img-thumbnail	Adds a border + padding to look like a photo frame.	Thumbnail-style border

Example: Rounded

```
<body>

</body>
```

Output:



Example: Circle

```
<body>

</body>
```

Output:



Thumbnail

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

Example

```
<body>

</body>
```

Output:



Note: Effect of **class= “img-fluid”** in **** - The responsive image is used to adjust the image automatically to the specified box

Alignment Classes (*Ref)

- **float-start**– Attaches image to leftmost side
- **float-end** – Attaches image to rightmost side
- **mx-auto d-block** – margin auto sets image as centrally aligned element

Example:

```
<body>



</body>
```

Output:



Table

- ✓ **Bootstrap** provides a series of classes that can be used to apply various styling to the tables such as changing the heading appearance, making the rows stripped, adding or removing borders, making rows hoverable, etc.
- ✓ Bootstrap also provides classes for making tables responsive.

Simple Table:

- The *.table* class is used to create a simple Bootstrap table.
- This class name is used with the `<table>` tag to create a table.

Syntax: `<table class="table"> Table Contents... </table>`

Class	Description
Table (Must)	Adds basic Bootstrap table styling.
table-striped	Adds zebra-striped rows (alternating light/dark rows).
table-bordered	Adds borders to all table cells.
table-borderless	Removes all borders.
table-hover	Adds a highlight effect when hovering over rows.
table-sm	Makes table more compact with reduced padding.
table-dark	Applies a dark theme to the entire table.
table-light	Applies a light theme (usually default).
table-primary / secondary / success / danger / warning / info / dark / light	Applies contextual color to the entire row or cell .
table-responsive	Makes table horizontally scrollable on small screens.

The *.table* class adds basic styling to a table

```
<table class="table">
  <thead>
    <tr>
      <th>S. No.</th>
      <th>Name</th>
      <th>City</th>
      <th>Age</th>
    </tr>
  </thead>
```

Introduction to Bootstrap 5: Utilities & Core Components

```
<tbody>
<tr>
  <th>1</th>
  <td>A</td>
  <td>Ahmedabad</td>
  <td>22</td>
</tr>
<tr>
  <th>2</th>
  <td>B</td>
  <td>Rajkot</td>
  <td>17</td>
</tr>
<tr>
  <th>3</th>
  <td>C</td>
  <td>Surat</td>
  <td>25</td>
</tr>
</tbody>
</table>
```

S. No.	Name	City	Age
1	A	Ahmedabad	22
2	B	Rajkot	17
3	C	Surat	25

❖ Striped Rows

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

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

Add **.table-striped** class in above example.

S. No.	Name	City	Age
1	A	Ahmedabad	22
2	B	Rajkot	17
3	C	Surat	25

Introduction to Bootstrap 5: Utilities & Core Components

❖ Bordered Table

The `.table-bordered` class adds borders on all sides of the table and cells

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

Add `.table-bordered` class in above example.

S. No.	Name	City	Age
1	A	Ahmedabad	22
2	B	Rajkot	17
3	C	Surat	25

❖ Hover Rows

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

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

Add `.table-hover` class in above example.

S. No.	Name	City	Age
1	A	Ahmedabad	22
2	B	Rajkot	17
3	C	Surat	25

```
<table class="table table-hover table-striped"> content </table>
```

Add `.table-hover` and `.table-striped` classes in above example.

S. No.	Name	City	Age
1	A	Ahmedabad	22
2	B	Rajkot	17
3	C	Surat	25

Introduction to Bootstrap 5: Utilities & Core Components

❖ Black/Dark Table

The `.table-dark` class adds a black background to the table

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

Add `.table-dark` class in above example.

S. No.	Name	City	Age
1	A	Ahmedabad	22
2	B	Rajkot	17
3	C	Surat	25

❖ Dark Striped Table

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

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

Add `.table-hover` and `.table-striped` classes in above example.

S. No.	Name	City	Age
1	A	Ahmedabad	22
2	B	Rajkot	17
3	C	Surat	25

❖ Hoverable Dark Table

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

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

Add `.table-hover` and `.table-dark` classes in above example.

Introduction to Bootstrap 5: Utilities & Core Components

S. No.	Name	City	Age
1	A	Ahmedabad	22
2	B	Rajkot	17
3	C	Surat	25

❖ Borderless Table

The `.table-borderless` class removes borders from the table

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

Add `.table-borderless` class in above example.

S. No.	Name	City	Age
1	A	Ahmedabad	22
2	B	Rajkot	17
3	C	Surat	25

❖ Contextual Classes

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

The contextual classes that can be used are:

Class Name	Indicates
.table-primary	Blue: Indicates an important action
.table-success	Green: Indicates a successful or positive action
.table-danger	Red: Indicates a dangerous or potentially negative action
.table-info	Light blue: Indicates a neutral informative change or action
.table-warning	Orange: Indicates a warning that might need attention
.table-secondary	Grey: Indicates a slightly less important action
.table-light	Light grey table or table row background
.table-dark	Dark grey table or table row background

```
<table class="table table-danger table-striped"> content </table>
```

Add **.table-danger** and **.table-striped** classes in above example.

Also, we can apply color to particular row or column by adding class to **<tr>** or **<td>** element.

S. No.	Name	City	Age
1	A	Ahmedabad	22
2	B	Rajkot	17
3	C	Surat	25

❖ Small table

The **.table-sm** class makes the table smaller by cutting cell padding in half.

```
<table class="table table-sm table-danger table-striped "> content </table>
```

Add **.table-borderless** class in above example.

S. No.	Name	City	Age
1	A	Ahmedabad	22
2	B	Rajkot	17
3	C	Surat	25

❖ Responsive Tables

- ✓ The **.table-responsive** class adds a scrollbar to the table when needed (when it is too big horizontally):
- ✓ You can also decide when the table should get a scrollbar, depending on screen width:

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

...

If you don't add table-responsive class then it will add horizontal scrollbar to whole web page, which is not correct way.

To make a table responsive add table-responsive class before the table tag starts (which makes only table responsive by adding horizontal scrollbar to table).

Introduction to Bootstrap 5: Utilities & Core Components

Flex

The biggest difference between Bootstrap 3 and Bootstrap 4 & 5 is that Bootstrap 5 now uses flexbox, instead of floats, to handle the layout.

The Flexible Box Layout Module, makes it easier to design flexible responsive layout structure without using float or positioning.

Class	Description
d-flex	Makes the element a flex container (display: flex).
d-inline-flex	Makes the element an inline flex container (display: inline-flex).
flex-row	Flex items are laid out in a row (horizontal).
flex-row-reverse	Flex items laid out in a row but in reverse order.
flex-column	Flex items laid out in a column (vertical).
flex-column-reverse	Flex items laid out in a column in reverse order.
justify-content-start justify-content-end justify-content-center justify-content-between justify-content-around justify-content-evenly	Controls alignment of flex items along main axis (horizontal for row, vertical for column).
align-items-start align-items-end align-items-center	Controls alignment of flex items along the cross axis.
flex-wrap flexnowrap flex-wrap-reverse	Allows flex items to wrap onto new lines when needed. Prevents wrapping; all items stay on a single line. Items wrap, but in reverse order (from bottom to top).

Example:

```
<div class="d-flex flex-row justify-content-around align-items-center border border-5 border-danger p-5" style="height:150px;">
  <div class="p-5 bg-primary text-white">Item 1</div>
  <div class="p-5 bg-danger text-white">Item 2</div>
  <div class="p-5 bg-success text-white">Item 3</div>
</div>
```



Basic Flex

d-flex - To create a flexbox container and to transform direct children into flex items.

Example:

```
<div class="container mt-3">
<h3>Flex</h3>
<div class="d-flex p-3 bg-secondary text-white">
  <div class="p-2 bg-info">
    ITEM-1
  </div>
  <div class="p-2 bg-warning">
    ITEM-2
  </div>
  <div class="p-2 bg-danger">
    ITEM-3
  </div>
</div>
</div>
```

Output:

Flex



Inline Flex

- **d-inline-flex** : To create an inline flexbox container

Modify your `<div class="d-inline-flex p-3 bg-secondary text-white">`

Output:

Flex



Horizontal Direction

- **.flex-row** : To display the flex items horizontally (side by side). This is default.
- **.flex-row-reverse** : To right-align the horizontal reverse direction (RTL)

Modify your `<div class="d-flex flex-row-reverse p-3 bg-secondary text-white">`

Output:

Flex



gVertical Direction

- **.flex-column:** To display the flex items vertically (on top of each other)

Modify your <div class="**d-flex flex-column p-3 bg-secondary text-white**">

Output:

Flex



- **.flex-column-reverse:** To reverse the vertical direction

Modify your <div class="**d-flex flex-column-reverse p-3 bg-secondary text-white**">

Output:

Flex



Justify

justify-content-* classes to change the alignment of flex items. Valid classes are start (default), end, center, between or around.

*add class with d-flex

```
<div class="d-flex justify-content-start p-3 bg-secondary text-white">
<div class="d-flex justify-content-end p-3 bg-secondary text-white">
<div class="d-flex justify-content-center p-3 bg-secondary text-white">
<div class="d-flex justify-content-between p-3 bg-secondary text-white">
<div class="d-flex justify-content-around p-3 bg-secondary text-white">
```

Output:

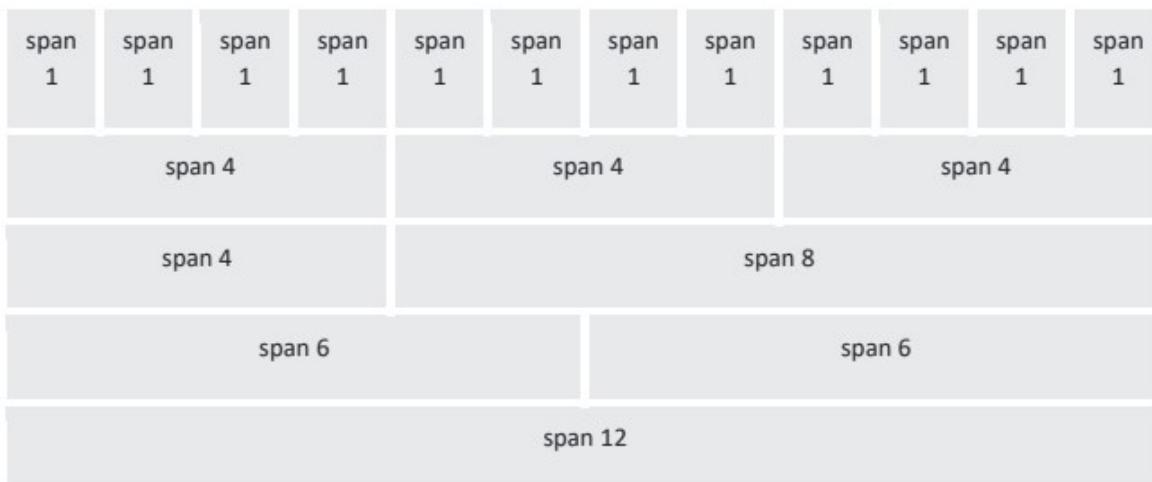


Introduction to Bootstrap 5: Utilities & Core Components

Grid System

- Bootstrap's grid system is built with flexbox and allows up to 12 columns across the page.
- If you do not want to use all 12 columns individually, you can group the columns together to create wider columns:
- The grid system is responsive, and the columns will re-arrange automatically depending on the screen size.
- Make sure that the sum adds up to 12 or fewer (it is not required that you use all 12 available columns).

Grid Classes



Breakpoint	Class Prefix	Width
Extra small	.col-	<576px
Small	.col-sm-	≥576px
Medium	.col-md-	≥768px
Large	.col-lg-	≥992px
Extra large	.col-xl-	≥1200px
Extra extra large	.col-xxl-	≥1400px

Class Pattern	Meaning	Behavior
col	Auto column	All .col in a row share equal width on all screens.
col-*	Fixed width	* = number (1-12); width stays same on all screens.
col-sm-*	Small screens up	Applies width at ≥576px; below this size it stacks. * = number (1-12);
col-md-*	Medium screens up	Applies width at ≥768px; below this it stacks. * = number (1-12);
col-lg- *	Large screens up	Applies width at ≥992px; below this it stacks.

Introduction to Bootstrap 5: Utilities & Core Components

		* = number (1-12);
col- <i>xl</i> *	Extra-large screens up	Applies width at $\geq 1200\text{px}$. * = number (1-12);
col- <i>xxl</i> *	Extra-extra-large screens up	Applies width at $\geq 1400\text{px}$. * = number (1-12);

class= “row” : A “.row” divides the page horizontally so you can place multiple .col-* elements inside it.

Class= “col-*-*” : It is a Bootstrap grid column class used to control how wide a column should be at different screen sizes.

col- → column

* (first star) → **breakpoint / screen size**

* (second star) → **column width (1 to 12)**

class: col-{breakpoint}-{width}

Tip: Each class scales up, so if you wish to set the same widths for sm and md, you only need to specify sm.

The following is a basic structure of a Bootstrap 4 grid:

```
<!-- Control the column width, and how they should appear on different devices -->
```

```
<div class="row">
  <div class="col-*-*"></div>
  <div class="col-*-*"></div>
  <div class="col-*-*"></div>
</div>
```

```
<!-- Or let Bootstrap automatically handle the layout -->
```

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

- **First example:** create a row (<div class="row">).
Then, add the desired number of columns (tags with appropriate .col-*-* classes).
The first star (*) represents the responsiveness: sm, md, lg or xl, while the second star represents a number, which should add up to 12 for each row.
- **Second example:** instead of adding a number to each col, let bootstrap handle the layout, three cols = 33.33% width to each col. You can also use .col-sm|md|lg|x1 to make the columns responsive.

Introduction to Bootstrap 5: Utilities & Core Components

Examples

1) .col (auto equal columns)

```
<div class="row">
  <div class="col">A</div>
  <div class="col">B</div>
  <div class="col">C</div>
</div>
```

→ 3 equal columns on all devices

2) .col-* (fixed width everywhere)

```
<div class="row">
  <div class="col-3">A</div>
  <div class="col-3">B</div>
  <div class="col-3">C</div>
  <div class="col-3">D</div>
</div>
```

→ Always 4 equal columns on all devices

3) .col-sm-* (responsive at 576px+)

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

- <576px → stacked
- ≥576px → 4 + 8 layout

Responsive Columns in bootstrap

- Bootstrap allows you to control **how wide a column should be** on different screen sizes.

Example: **col-12 col-sm-6 col-md-4 col-lg-3**

- On **mobile** → full width
- On **small screens** → half
- On **medium screens** → one-third
- On **large screens** → one-fourth

```
<div class="container mt-4">
  <div class="row">
    <div class="col-12 col-sm-6 col-md-4 col-lg-3 bg-primary text-white p-3 mb-3">
      Box 1
    </div>
    <div class="col-12 col-sm-6 col-md-4 col-lg-3 bg-success text-white p-3 mb-3">
      Box 2
    </div>
    <div class="col-12 col-sm-6 col-md-4 col-lg-3 bg-warning text-dark p-3 mb-3">
      Box 3
    </div>
  </div>
</div>
```

Introduction to Bootstrap 5: Utilities & Core Components

```
</div>

<div class="col-12 col-sm-6 col-md-4 col-lg-3 bg-danger text-white p-3 mb-3">
  Box 4
</div>
</div>
</div>
```

📱 On Mobile (<576px)

All boxes take **full width**, so they appear **stacked vertically**.



⌚ On Tablets (≥576px)

Two boxes appear per row.



💻 On Medium Devices (≥768px)

Three boxes per row.



📺 On Large Screens (≥992px)

Four boxes per row (like a typical gallery layout).



Note: Always include `<meta name="viewport" content="width=device-width, initial-scale=1">` to ensure Bootstrap 5 responsive classes like `col-sm`, `col-md`, and `col-lg` work correctly.

Introduction to Bootstrap 5: Utilities & Core Components

Buttons

- Bootstrap provides different classes that can be used with different tags, such as <button>, <a>, <input> to apply custom button styles.
- Bootstrap also provides classes that can be used for changing the state and size of buttons.
- Bootstrap contains many classes to set the style of the button element. The list of button classes are given below:

- .btn
- .btn-primary
- .btn-secondary
- .btn-success
- .btn-info
- .btn-warning
- .btn-danger
- .btn-dark
- .btn-light
- .btn-link

Basic Primary Secondary Success Info Warning Danger Dark Light Link

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

The button classes can be used on <a>, <button>, or <input> elements:

Example:

```
<div class="container">
  <a href="#" class="btn btn-warning" role="button">Link Button</a>
  <button type="button" class="btn btn-danger">Button</button>
  <input type="button" class="btn btn-primary" value="Input Button">
  <input type="submit" class="btn btn-secondary" value="Submit Button">
</div>
```

Link Button Button Input Button Submit Button

Introduction to Bootstrap 5: Utilities & Core Components

put a # in the href attribute of the link?

Since we do not have any page to link it to, and we do not want to get a "404" message, we put # as the link. In real life it should of course been a real URL to the "Search" page.

Button Outline

Bootstrap provides eight outline/bordered buttons:

- ✓ Primary
- ✓ Secondary
- ✓ Success
- ✓ Info
- ✓ Warning
- ✓ Danger
- ✓ Dark
- ✓ Light



Example

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

Hover effect on btn-danger



Button Sizes

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

- Large
- Default
- Small



Introduction to Bootstrap 5: Utilities & Core Components

Example

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

Block Level Buttons

Add class **.w-100** to create a block level button that spans the entire width of the parent element.

Example

```
<button type="button" class="btn btn-primary w-100 ">Full-Width Button</button>
```



Full-Width Button

Active/Disabled Buttons

A button can be set to an active (appear pressed) or a disabled (unclickable) state:

- ✓ The class **.active** makes a button appear pressed
- ✓ The **disabled** attribute makes a button unclickable.

Note that **<a>** elements do not support the **disabled** attribute and must use the **.disabled** class to make it visually appear disabled.

Example

```
<div class="container">
  <h2>Button States</h2>
  <button type="button" class="btn btn-danger">Primary Button</button>
  <button type="button" class="btn btn-danger active">Active Primary</button>
  <button type="button" class="btn btn-danger" disabled>Disabled Primary</button>
  /*disabled attribute for button*/
  <a href="#" class="btn btn-danger disabled">Disabled Link</a> /*disabled class for link*/
</div>
```

Button States

Primary Button Active Primary Disabled Primary Disabled Link

Spinners

The Spinners in Bootstrap can be used to indicate the loading state in the web page. Bootstrap also provides various classes for creating different styles of spinners.

The list of those various styles of Spinners:

1. **Bordered Spinner**
2. **Colored Spinners**
3. **Growing Spinners**
4. **Spinner Size**
5. **Buttons with Spinners**

1. Border Spinner:

A user can create bordered spinner with the help of **.spinner-border** class.



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

2. Colored Spinners:

Use any **text color utilities** to add a color to the spinner:



```
<div class="spinner-border text-muted"></div>
<div class="spinner-border text-primary"></div>
<div class="spinner-border text-success"></div>
<div class="spinner-border text-info"></div>
<div class="spinner-border text-warning"></div>
<div class="spinner-border text-danger"></div>
<div class="spinner-border text-secondary"></div>
<div class="spinner-border text-dark"></div>
<div class="spinner-border text-light"></div>
```

3. Growing Spinners:

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



Introduction to Bootstrap 5: Utilities & Core Components

```
<div class="spinner-grow text-muted"></div>
<div class="spinner-grow text-primary"></div>
<div class="spinner-grow text-success"></div>
<div class="spinner-grow text-info"></div>
<div class="spinner-grow text-warning"></div>
<div class="spinner-grow text-danger"></div>
<div class="spinner-grow text-secondary"></div>
<div class="spinner-grow text-dark"></div>
<div class="spinner-grow text-light"></div>
```

4. Spinner Size

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



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

5. Spinner Buttons

We can also add "spinners" to a button.



Example

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

Alerts

- We often see certain **alerts** on some websites before or after completing an action.
- These alert messages are highlighted texts that are important to take into consideration while performing a process.
- Bootstrap allows showing these alert messages on the website using predefined classes.
- The **.alert** class followed by contextual classes are used to display the alert message on website.
- The alert classes are:
 - **.alert-success,**
 - **.alert-info,**
 - **.alert-warning,**
 - **.alert-danger,**
 - **.alert-primary,**
 - **.alert-secondary,**
 - **.alert-light**
 - **.alert-dark.**

Syntax:

```
<div class="alert alert-success"> Contents... </div>
```

Part	Meaning	Explanation
alert	Makes the element a Bootstrap alert	Applies alert formatting
.alert-success, .alert-info, .alert-warning, .alert-danger, .alert-primary, .alert-secondary, .alert-light .alert-dark.	Color of an alert	Contextual color
alert-dismissible	Makes the alert closable	Adds padding for the close button
fade	Adds fade animation	Alert fades out smoothly when closed
show	Makes fade animation visible	Shows alert immediately with fade effect
btn-close	Bootstrap close (x) button	Replaces the old close icon
data-bs-dismiss="alert"	Dismisses (closes) the alert	Tells Bootstrap JS to remove the alert when clicked

Introduction to Bootstrap 5: Utilities & Core Components

Example:

```
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<!-- Bootstrap 5 CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>

<div class="alert alert-warning alert-dismissible fade show">
  Important message
  <button type="button" class="btn-close" data-bs-dismiss="alert" ></button>
</div>

<!-- Bootstrap JS -->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</body>
```

Important message



Dropdowns

- **Dropdowns** are one of the most important parts of an interactive website.
- A dropdown menu is the collection of menu items that allow users to choose a value from the list.
- The `.dropdown` class is used to design the drop-down menu.

Class / Attribute	Type	Meaning / Use
dropdown	Class	Wrapper that activates dropdown component.
btn	Class	Base button styling.
btn-secondary	Class	Button color style (secondary theme).
dropdown-toggle	Class	Adds dropdown arrow and enables toggle styling.
dropdown-menu	Class	The menu container that appears when opened.
dropdown-item	Class	Styles each menu link/item.
data-bs-toggle="dropdown"	Attribute	Most important attribute. Tells Bootstrap to open/close the dropdown.
dropdown-header	Class	Creates a non-clickable header inside dropdown.
dropdown-divider	Class	Creates a horizontal separator line.
disabled	Class	Makes a dropdown item unclickable and faded.
active	Class	Highlights the dropdown item as selected.

Example:

```

<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<!-- Bootstrap 5 CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
<div class="dropdown">
<button class="btn btn-secondary dropdown-toggle" data-bs-toggle="dropdown" >
    Dropdown button
</button>
<ul class="dropdown-menu">
    <li><a class="dropdown-item" href="#">test 1</a></li>
    <li><a class="dropdown-item" href="#">test 2</a></li>

```

Introduction to Bootstrap 5: Utilities & Core Components

```
<li><a class="dropdown-item" href="#">test 3</a></li>
</ul>
</div>
<!-- Bootstrap JS -->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</body>
```



Example with Dropdown Divider, Dropdown Header, Disable and Active items:

```
<div class="container m-4">
<div class="dropdown">
<button class="btn btn-success dropdown-toggle" data-bs-toggle="dropdown">
    Subjects
</button>
<div class="dropdown-menu">
    <strong class="dropdown-header">Sem 3</strong>
    <a class="dropdown-item active" href="#">FSD-1</a>
    <a class="dropdown-item disabled" href="#">FCSP-1</a>
    <a class="dropdown-item" href="#">DE</a>
    <a class="dropdown-item" href="#">PS</a>

    <div class="dropdown-divider"></div>

    <strong class="dropdown-header">Sem 4</strong>
    <a class="dropdown-item" href="#">FSD-2</a>
    <a class="dropdown-item disabled" href="#">FCSP-2</a>
    <a class="dropdown-item" href="#">TOC</a>
</div>
</div>
</div>
```



Cards

Bootstrap Cards are flexible, responsive content containers used to display information neatly. A card can include text, images, buttons, lists, and links, all styled with a lightweight and consistent design. Cards support headers, footers, grids, and various layouts, making them ideal for product displays, profiles, articles, and UI components.

Class Name	Purpose / Description
card	The main container that creates a Bootstrap card.
card-img-top	It rounds the top corners automatically, Places an image at the top of the card. It automatically sizes the image to fit the card's width.
card-body	A section inside the card that holds text, links, buttons, and other content. Adds padding.
card-title	Styles the title text inside the card body. Usually used with headings (h5, h4, etc.).
card-text	Styles regular paragraph text inside the card body.
card-header	Adds a styled header section to the top of the card. <i>Optional.</i>
card-link	Styles links inside cards and spaces them properly.

Example

```
<div class="card w-25">
  <div class="card-header text-danger">
    Sale
  </div>
  
  <div class="card-body">
    <h5 class="card-title">Card title</h5>
    <p class="card-text">Some random content</p>
    <a href="#" class="btn btn-danger">Go somewhere</a>
  </div>
</div>
```



Collapse

- Collapsible is useful when you want to hide and show large amount of content.
- Collapse component is used to show or hide the content by using `.collapse` class.
- The content can be collapsed by adding `data-bs-toggle="collapse"` attribute anchor or button element. The id of these elements references to the id of the content to collapse the data.

Item	Type	Meaning / Function
<code>btn</code>	Class	Base Bootstrap button styling.
<code>btn-primary</code>	Class	Blue primary button style.
<code>data-bs-toggle="collapse"</code>	Attribute	Tells Bootstrap JS to toggle a collapse component when clicked.
<code>href="#collapseExample"</code>	Attribute (link version)	The target collapse element ID (for <code><a></code> tag triggers). Works the same as <code>data-bs-target</code> .
<code>data-bs-target="#collapseExample"</code>	Attribute (button version)	Points to the collapse element by ID (used with <code><button></code>).
<code>collapse</code>	Class	Makes the div collapsible (hidden by default).
<code>id="collapseExample"</code>	Attribute	Matches the trigger's href or <code>data-bs-target</code> .
<code>card-body</code>	Class	Bootstrap card component wrapper.
<code>card</code>	Class	Proper padding and text formatting inside card.
<code>show</code>	Class	Make the collapsible element visible

Example:

```

<div>
<a class="btn btn-primary" data-bs-toggle="collapse" href="#collapseExample" >
    Link with href
</a>
<div class="collapse" id="collapseExample">
    <div class="card card-body">
        Some placeholder content for the collapse component. This panel is hidden by default
        but revealed when the user activates the relevant trigger.
    </div>
</div>
</div>

<div>
    <button data-bs-toggle="collapse" data-bs-target="#collapseExample1" class="btn btn-primary" >
        Button with data-bs-target
    </button>
</div>

```

Introduction to Bootstrap 5: Utilities & Core Components

```
</button>
<div class="collapse" id="collapseExample1">
  <div class="card card-body">
    Some placeholder content for the collapse component. This panel is hidden by default
    but revealed when the user activates the relevant trigger.
  </div>
</div>
</div>
```

Link with href

Button with data-bs-target

Some placeholder content for the collapse component. This panel is hidden by default but revealed when the user activates the relevant trigger.

The class **show** is used with **.collapse** to make the content visible by default.

```
<div class="collapse show" id="collapseExample"></div>
```

Accordion

- Accordion is a graphical element consisting of a vertically stacked list of items.
- This can be clicked to reveal or hide content associated with them.
- In this digital era, every website contains an accordion at least one of their pages if they have a FAQ section.
- With the use of accordion you can make your content display more interactive.
- The use of the data-bs-parent attribute to makes sure that all collapsible elements under the specified parent will be closed when one of the collapsible items is display.

Class / Attribute	Used On	Meaning / Purpose
Container-Level Classes		
accordion	<div id="accordionExample">	Creates a Bootstrap accordion wrapper.
accordion-item	Each accordion section	Defines a single accordion item/block.
Header & Button Classes		
accordion-header	<h2>	Wrapper for accordion button (not clickable itself).
accordion-button	<button>	Styled button that triggers expand/collapse.
collapsed	<button> (when closed)	Indicates collapsed state (Bootstrap adds/removes this).
data-bs-toggle="collapse"	<button>	Tells Bootstrap to toggle collapse behavior.
data-bs-target="#collapseOne"	<button>	Points to the element that will collapse/expand.
Collapsible Content Classes		
accordion-collapse	<div>	Wrapper for collapsible content inside accordion.
collapse	<div>	Makes the content collapsible (hidden by default).
show	<div>	Makes this section visible by default.
id="collapseOne"	<div>	Unique ID used to match a button's data-bs-target.
data-bs-parent="#accordionExample"	<div>	Ensures only one accordion item opens at a time (auto-close others).
Content Body Classes		
accordion-body	<div>	Provides padding, text formatting, and structure for the content inside each accordion panel.

Example:

```
<div class="accordion" id="parent">

  <div class="accordion-item">
    <h2 class="accordion-header">
      <button class="accordion-button" data-bs-toggle="collapse" data-bs-target="#collapseOne">
        Accordion Item #1
      </button>
    </h2>
    <div id="collapseOne" class="accordion-collapse collapse show" data-bs-parent="#parent">
      <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.
      </div>
    </div>
  </div>

  <div class="accordion-item">
    <h2 class="accordion-header">
      <button class="accordion-button collapsed" data-bs-toggle="collapse" data-bs-target="#collapseTwo">
        Accordion Item #2
      </button>
    </h2>
    <div id="collapseTwo" class="accordion-collapse collapse" data-bs-parent="#parent">
      <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.
      </div>
    </div>
  </div>

  <div class="accordion-item">
    <h2 class="accordion-header">
      <button class="accordion-button collapsed" data-bs-toggle="collapse" data-bs-target="#collapseThree">
        Accordion Item #3
      </button>
    </h2>
    <div id="collapseThree" class="accordion-collapse collapse" data-bs-parent="#parent">
      <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.
      </div>
    </div>
  </div>
</div>
```

Introduction to Bootstrap 5: Utilities & Core Components

Accordion Item #1

This is the first item's accordion body. It is shown by default, until the collapse plugin adds the appropriate classes that we use to style each element.

Accordion Item #2

Accordion Item #3

Bootstrap Icons

What Are Bootstrap Icons?

Bootstrap Icons are a free collection of more than 2,000 icons designed to work with Bootstrap 5. They represent common user interface elements such as home, search, download, edit, and more.

Note: Bootstrap itself does not include icons. Bootstrap Icons is a separate library that must be added manually.

2. Why Use Bootstrap Icons?

1. Makes the user interface clearer and easier to understand
Icons help users identify actions quickly.
2. Provides a modern and consistent look
The icons match the design style of Bootstrap.
3. Easy to use
You only need one CDN link and a simple HTML tag to display an icon.
4. Lightweight
Icons load quickly on all devices.
5. Customizable
You can change the icon's size, color, and alignment using CSS.

3. Where to Find Icon Names

All icons and their class names can be found on the official Bootstrap Icons website:
<https://icons.getbootstrap.com>

4. How to Add Bootstrap Icons (CDN Link)

Add the following CDN link inside the <head> section of your HTML file:

```
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstrap-icons.css">
```

After adding this link, you can use any icon from the Bootstrap Icons library.

5. How to Use Bootstrap Icons

To display an icon, use the `<i>` tag (or ``) and apply the correct icon class name.

Basic syntax:

```
<i class="bi bi-star"></i>
```

Example:

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <!-- Bootstrap 5 CSS -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstrap-icons.css">

</head>
<body>
  <i class="bi bi-star" ></i>
  <br>
  <i class="bi bi-person" ></i> Profile
  <br>
  <button class="btn btn-primary">
    <i class="bi bi-download" style="font-size: 1.5em; vertical-align: middle;"></i> Download
  </button>
</body>
```

