

# Chapter 6

## What is Bootstrap?

- Bootstrap is the most popular **HTML, CSS, and JavaScript framework** for developing **responsive, mobile-friendly websites**.
- It is **free** to download and use.
- It is a **front-end framework** designed to make web development **easier, faster, and consistent**.
- Includes ready-made templates for:
  - Typography
  - Forms
  - Buttons
  - Tables
  - Navigation bars
  - Modals
  - Carousels and many more UI components.
- Helps create **responsive designs** with minimal effort.

## What is a 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**.



## History of Bootstrap

- Developed by **Mark Otto** and **Jacob Thornton** at **Twitter**.
- Released as an **open-source** framework in **August 2011** on GitHub.

## Why Use Bootstrap?

- **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**.

- **Browser compatibility:** Works with all major browsers:
  - Chrome
  - Firefox
  - Safari
  - Opera
  - Edge
  - Internet Explorer

## Where to Get Bootstrap?

You can start using Bootstrap in two ways:

### 1. Use Bootstrap via CDN

### 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.

#### Just for knowledge:

##### 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>&lt;link rel="stylesheet" href="bootstrap.min.css"&gt;</code>	Loads Bootstrap's styles (layout, components, utilities). <b>Must be included in every Bootstrap project.</b>	Always use. Needed for all Bootstrap designs and components.
Bundle JS (Recommended)	<code>&lt;script src="bootstrap.bundle.min.js"&gt;&lt;/script&gt;</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>&lt;script src="popper.min.js"&gt;&lt;/script&gt; &lt;script src="bootstrap.min.js"&gt;&lt;/script&gt;</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/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTWfspd3yD65VohhpuuCOmLASjC" crossorigin="anonymous">
```

## Bundle JS (Recommended)

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
integrity="sha384-MrcW6ZMFYlzcLA8Nl+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-IQsoLX15PILFhosVNubq5LC7Qb9DXgDA9i+tQ8Zj3iwWAwPtgFTxbJ8NT4GN1R8p"
crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.min.js"
integrity="sha384-cVKIPhGWiC2A14u+LWgxfKTRICfu0JtXr+EQDz/bglDoEy14H0zUF0QKbrJ0EcQF"
crossorigin="anonymous"></script>
```

### Just for knowledge:

#### 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-
sRII4kxILFvY47J16cr9ZwB07vP4J8+LH7qKQnuqkuIAvNWLzeN8tE5YBujZqJLB"
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-
FKyoeEForCGlyvw9Hj09JcYn3nv7wiPvlz7YYwJrWVcXK/BmnVDxM+D2scQbITxl"
crossorigin="anonymous"></script>
</body>
</html>
```

### Just for knowledge:

#### Where to put the <link> tag?

- 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.**

#### Where to put the <script> tag?

- 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](https://getbootstrap.com)
- You can host the files locally.

### How to Download Bootstrap

1. Go to [getbootstrap.com](https://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**

## container vs container-fluid

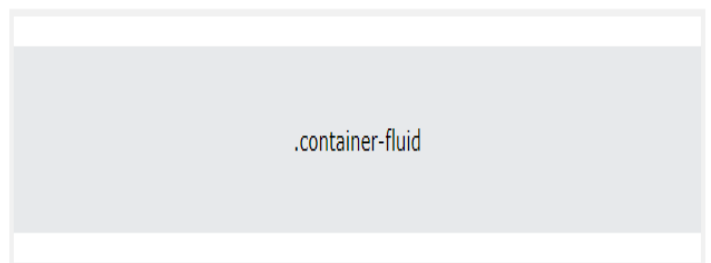
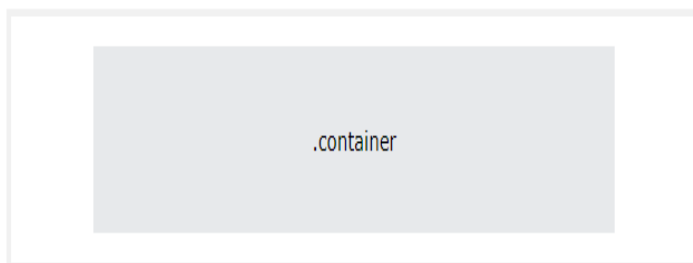
### .container (fixed width)

- A container keeps your content centered and prevents it from becoming too wide. It gives your page a clean, readable layout and is used as the base for the grid system.

### .container-fluid (full width)

- A container-fluid makes your content stretch across the entire screen. 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 <b>fixed width</b> and stays <b>centered</b> on the page.
.container-fluid	Always <b>100% full width</b> , stretching across the entire screen.



## Bootstrap 5 Breakpoints & Container Max-Width

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

### How .container-\* Classes Work

#### 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 (≥ 768px) and up.

## Bootstrap Utility Classes

### Spacing Utilities

#### 1. Property (what you are setting)

Token	Meaning
<b>m</b>	margin
<b>p</b>	padding

#### 2. Side (where it applies)

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

#### 3. Size (how much spacing)

Token	Value (default)
<b>0</b>	0
<b>1</b>	0.25rem
<b>2</b>	0.5rem
<b>3</b>	1rem
<b>4</b>	1.5rem
<b>5</b>	3rem



Token	Value (default)
<b>auto</b>	auto (margin only)

#### Examples: {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

## Sizing

### Width Classes (w-\*)

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

### Height Classes (h-\*)

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

## Text Colors

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

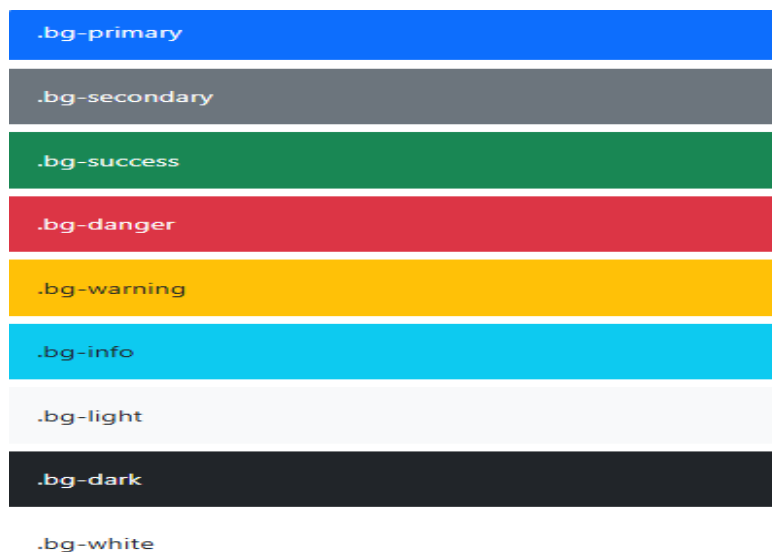
This text is light grey (on white background).

## Background Colors

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

### Example:

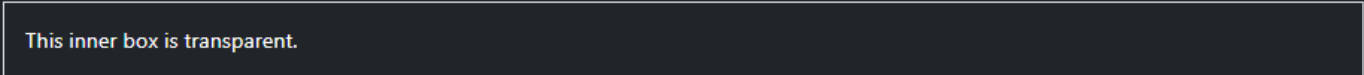
```
<div class="p-3 mb-2 bg-primary text-white">.bg-primary</div>
<div class="p-3 mb-2 bg-secondary text-white">.bg-secondary</div>
<div class="p-3 mb-2 bg-success text-white">.bg-success</div>
<div class="p-3 mb-2 bg-danger text-white">.bg-danger</div>
<div class="p-3 mb-2 bg-warning text-dark">.bg-warning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




## Understand the concept 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>
```



This inner box is transparent.



This inner box is not transparent.

## 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 <b>first letter of each word</b> , 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 <b>small, subtle shadow</b> , giving a light lifted look without strong depth.
shadow	Applies a <b>standard (medium) shadow</b> , commonly used for cards and floating elements.
shadow-lg	Adds a <b>large, deeper shadow</b> to create strong elevation and highlight prominence.

## Borders

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

## Example

```
<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>
```



## Display

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

## Positioning

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



Class	Detailed Description
<b>bottom-0 / bottom-50 / bottom-100</b>	Moves the element <b>upward</b> from the bottom by the given percentage.
<b>start-0 / start-50 / start-100</b>	Offsets from the <b>start side</b> (left in LTR, right in RTL).
<b>end-0 / end-50 / end-100</b>	Offsets from the <b>end side</b> (right in LTR, left in RTL).

## 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>

```

```
</div>  
</div>
```

```
<!-- Bootstrap JS -->
```

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

## BOOTSTRAP UTILITIES DEMO

### All Utilities In One Box

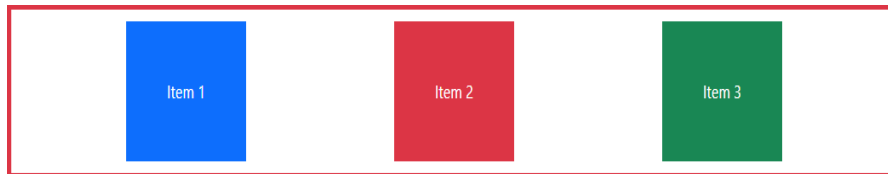
Lorem Ipsum Dolor Sit Amet Consectetur Adipiscing Elit. Accusamus Labore Quas,  
Beatae Delectus Nisi Dolorem Quod Neque Iste Ea Praesentium, Cum, Eligendi  
Amet Provident Culpa Ipsam Tenetur Repudiandae? Quod, Itaque.

## Flex

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.

### 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>
```



## 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	What It Looks Like
<b>img-fluid</b>	Makes image responsive.	Image adjusts size automatically
<b>rounded</b>	Gives the image small rounded corners.	Slightly curved edges
<b>rounded-circle</b>	Makes the image a perfect circle	Circular profile photo
<b>img-thumbnail</b>	Adds a border + padding to look like a photo frame.	Thumbnail-style border

### Example:

```

```

## Tables

- ✓ **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 striped, 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
<b>table</b>	Adds basic Bootstrap table styling.
<b>table-striped</b>	Adds zebra-striped rows (alternating light/dark rows).
<b>table-bordered</b>	Adds borders to all table cells.
<b>table-borderless</b>	Removes all borders.
<b>table-hover</b>	Adds a highlight effect when hovering over rows.
<b>table-sm</b>	Makes table more compact with reduced padding.
<b>table-dark</b>	Applies a dark theme to the entire table.
<b>table-light</b>	Applies a light theme (usually default).
<b>table-primary / secondary / success / danger / warning / info / dark / light</b>	Applies contextual color to the <b>entire row or cell</b> .
<b>table-responsive</b>	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>
<tbody>
    <tr>
        <th scope="row">1</th>
        <td>A</td>
        <td>Ahmedabad</td>
        <td>22</td>
    </tr>
    <tr>
        <th scope="row">2</th>
        <td>B</td>
        <td>Rajkot</td>
        <td>17</td>
    </tr>
    <tr>
        <th scope="row">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

### ❖ 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

### ❖ 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.

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:

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

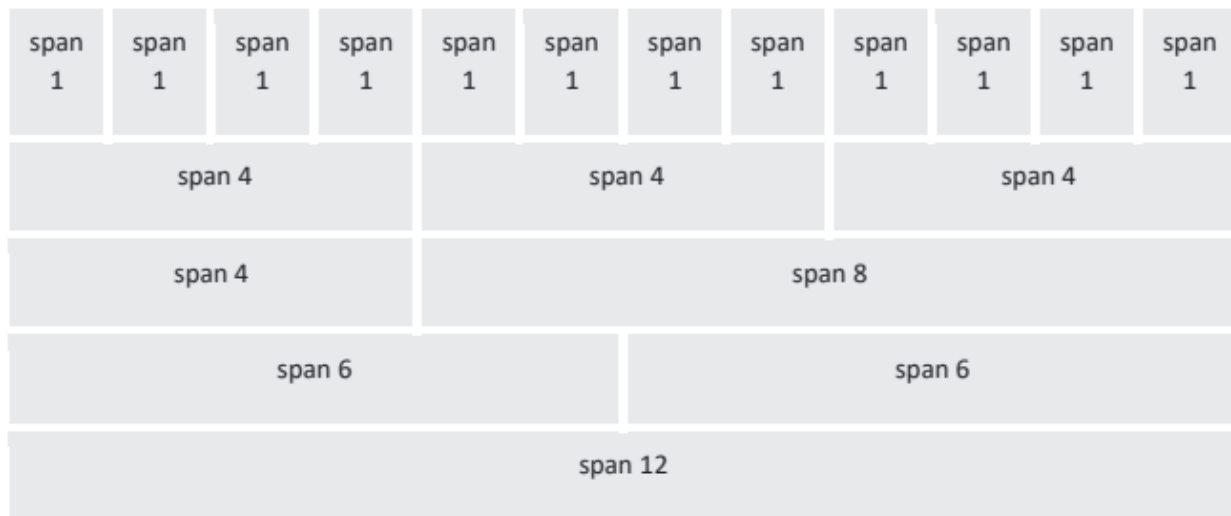
To make a table responsive add `table-responsive` class before the table tag starts (**which makes only table responsive by adding horizontal scrollbar to table**).

## Grid options

### 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
<b>col</b>	Auto column	All .col in a row share equal width on all screens.
<b>col-*</b>	Fixed width	* = number (1-12); width stays same on all screens.
<b>col-sm-*</b>	Small screens up	Applies width at ≥576px; below this size it stacks. * = number (1-12);
<b>col-md-*</b>	Medium screens up	Applies width at ≥768px; below this it stacks. * = number (1-12);
<b>col-lg-*</b>	Large screens up	Applies width at ≥992px; below this it stacks. * = number (1-12);
<b>col-xl-*</b>	Extra-large screens up	Applies width at ≥1200px. * = number (1-12);
<b>col-xxl-*</b>	Extra-extra-large screens up	Applies width at ≥1400px. * = number (1-12);

## 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 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>
```



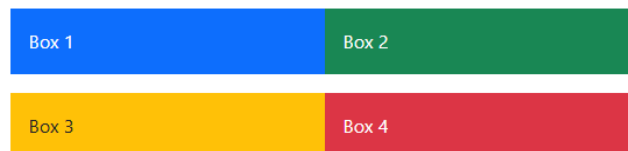
On Mobile (<576px)

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



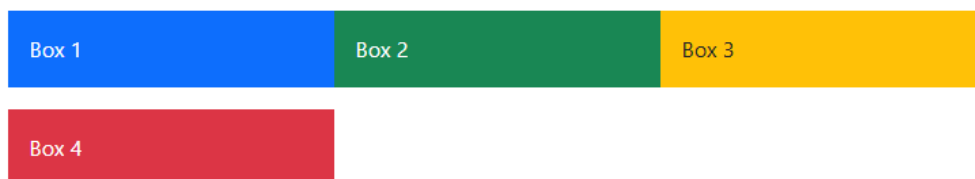
On Tablets (≥576px)

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).



## 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>
```

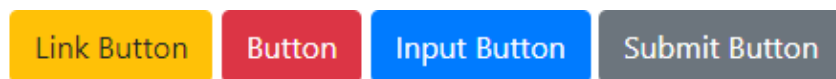
```
<button type="button" class="btn btn-link">Link</button>
```



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>
```



### Why do we 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>
```

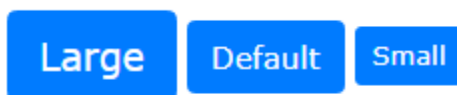
## Hover effect on btn-danger



## Button Sizes

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

- ✓ Large
- ✓ Default
- ✓ Small

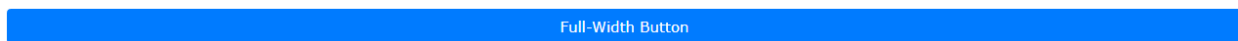


## Example

```
<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>
```

## 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":



```
<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

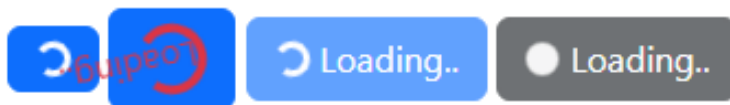
Use `.spinner-border-sm` or `.spinner-grow-sm` to create a smaller spinner:



```
<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:
  1. *.alert-success*,
  2. *.alert-info*,
  3. *.alert-warning*,
  4. *.alert-danger*,
  5. *.alert-primary*,
  6. *.alert-secondary*,
  7. *.alert-light*
  8. *.alert-dark*.

### Syntax:

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

Part	Meaning	Explanation
<b>alert</b>	Makes the element a Bootstrap alert	Applies alert formatting
<i>.alert-success</i> , <i>.alert-info</i> , <i>.alert-warning</i> , <i>.alert-danger</i> , <i>.alert-primary</i> , <i>.alert-secondary</i> , <i>.alert-light</i> <i>.alert-dark</i> .	Color of an alert	Contextual color
<b>alert-dismissible</b>	Makes the alert closable	Adds padding for the close button
<b>fade</b>	Adds fade animation	Alert fades out smoothly when closed
<b>show</b>	Makes fade animation visible	Shows alert immediately with fade effect
<b>btn-close</b>	Bootstrap close (×) button	Replaces the old close icon

Part	Meaning	Explanation
<b>data-bs-dismiss="alert"</b>	Dismisses (closes) the alert	Tells Bootstrap JS to remove the alert when clicked

### 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>

```





## Dropdowns

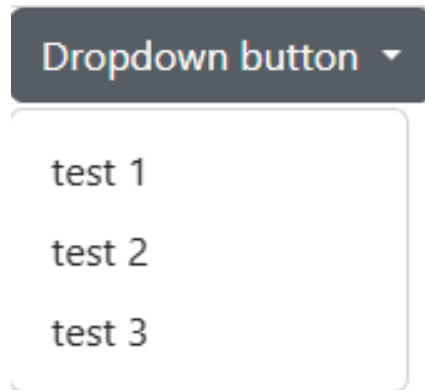
- ✓ **Dropdowns** are one of the most important parts of an interactive website.
- ✓ A dropdown menu is the collection of menu items that allow users to choose a value from the list.
- ✓ The `.dropdown` class is used to design the drop-down menu.

Class / Attribute	Type	Meaning / Use
<b>dropdown</b>	Class	Wrapper that activates dropdown component.
<b>btn</b>	Class	Base button styling.
<b>btn-secondary</b>	Class	Button color style (secondary theme).
<b>dropdown-toggle</b>	Class	Adds dropdown arrow and enables toggle styling.
<b>dropdown-menu</b>	Class	The menu container that appears when opened.
<b>dropdown-item</b>	Class	Styles each menu link/item.
<b>data-bs-toggle="dropdown"</b>	Attribute	<b>Most important attribute.</b> Tells Bootstrap to open/close the dropdown.

### 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>
    <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>
```



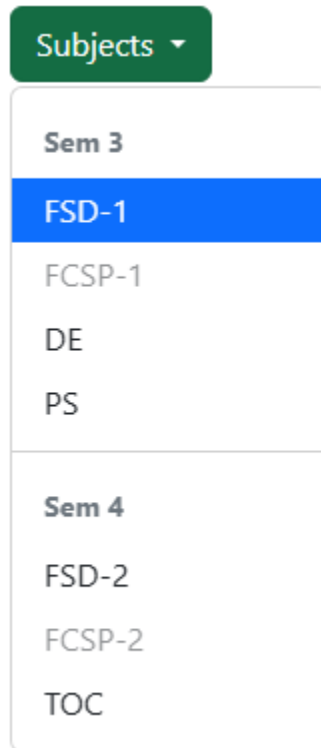
**Dropdown Divider:** The `.dropdown-divider` class is used to divide the dropdown menu list by using thin horizontal line.

**Dropdown Header:** The `.dropdown-header` class is used to add header section inside the dropdown list.

**Disable and Active items:** The `.active` class is used to add the highlight the list items. The `.disabled` class is used to disable the list of 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>  
    </div>  
  </div>  
</div>
```

```
<a class="dropdown-item disabled" href="#">FCSP-2</a>  
<a class="dropdown-item" href="#">TOC</a>  
</div>  
</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
btn	Class	Base Bootstrap button styling.
btn-primary	Class	Blue primary button style.
data-bs-toggle="collapse"	Attribute	Tells Bootstrap JS to toggle a collapse component when clicked.
href="#collapseExample"	Attribute (link version)	The target collapse element ID (for <a> tag triggers). Works the same as data-bs-target.
data-bs-target="#collapseExample"	Attribute (button version)	Points to the collapse element by ID (used with <button>).
collapse	Class	Makes the div collapsible (hidden by default).
id="collapseExample"	Attribute	Matches the trigger's href or data-bs-target.
card-body	Class	Bootstrap card component wrapper.
card	Class	Proper padding and text formatting inside card.

### 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 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.

### 1. Container-Level Classes

Class / Attribute	Used On	Meaning / Purpose
accordion	<div id="accordionExample">	Creates a Bootstrap accordion wrapper.
accordion-item	Each accordion section	Defines a single accordion item/block.

### 2. Header & Button Classes

Class / Attribute	Used On	Meaning / Purpose
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.

### 3. Collapsible Content Classes

Class / Attribute	Used On	Meaning / Purpose
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).

### 4. Content Body Classes

Class / Attribute	Used On	Meaning / Purpose
accordion-body	<div>	Provides padding, text formatting, and structure for the content inside each accordion panel.

#### Example:

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

  <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="#accordionExample">
```

```
<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="#accordionExample">
```

```
<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="#accordionExample">
```

```
<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>

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

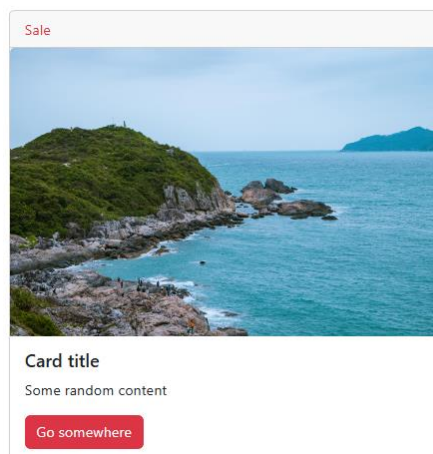


## Cards

Class Name	Purpose / Description
card	The main container that creates a Bootstrap card.
card-img-top	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. Optional.
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>
```



# 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. 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.

## 4. How to Use Bootstrap Icons

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

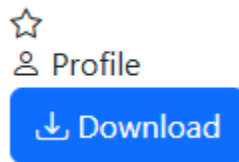
Basic syntax:

```
<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"></i> Download
  </button>
</body>
```



## 5. Where to Find Icon Names

All icons and their class names can be found on the official Bootstrap Icons website:

<https://icons.getbootstrap.com>

## Purpose of different attributes.

Attribute	Values / Examples	Purpose
<b>data-bs-toggle</b>	modal, collapse, dropdown, button	Defines which Bootstrap component behavior to activate.
<b>data-bs-target</b>	#elementId, .className	Specifies the element that will be opened/toggled by the trigger.
<b>data-bs-dismiss</b>	modal, alert	Closes/dismisses the specified component when clicked.
<b>data-bs-parent</b>	#accordionParent	Groups collapsible items so only one stays open (accordion behavior).