Bootstrap and Tailwind

By: Gino Fernando

What is Bootstrap?

Bootstrap is a free CSS framework. It helps you create good-looking, responsive websites quickly using pre-made styles and layouts.

Key Features:

- Mobile-friendly design
- Easy to learn
- Uses HTML + class names
- Built-in layout and components

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">

Bootstrap Grid System

Bootstrap's grid is a flexbox-based layout system that divides the page into a 12-column structure. You use it to arrange content into responsive rows and columns that adapt to screen sizes.

It's made up of:

- Containers (the outer wrapping elements)
- Rows (horizontal groups)
- Columns (the actual content areas)
- Breakpoints (screen size thresholds for responsive design)
- Column utilities (alignment, ordering, offsets, nesting)

Bootstrap Grid System

Bootstrap's grid is a flexbox-based layout system that divides the page into a 12-column structure. You use it to arrange content into responsive rows and columns that adapt to screen sizes.
It's made up of:

- Containers (the outer wrapping elements)
- Rows (horizontal groups)
- Columns (the actual content areas)
- Breakpoints (screen size thresholds for responsive design)
- Column utilities (alignment, ordering, offsets, nesting)

Container

Purpose

- A container is the outermost wrapper for your content.
- It centers and pads your content horizontally.
- Every Bootstrap layout should be inside a container to align properly with the grid.

Container

- Containers are the outermost wrapper for your grid.
- They control horizontal padding and alignment.
- Three main types:
  a. .container – fixed width that changes at breakpoints.
  b. .container-fluid – full-width (100%) at all screen sizes.
  c. .container-{breakpoint} – fluid until that breakpoint, then fixed.

Rows

- .row creates a horizontal group of columns.
- Uses negative margins to align columns with container padding.
- All columns must be placed inside a row.
-

```
<div class="row">
 <div class="col">Column 1</div>
 <div class="col">Column 2</div>
</div>
```

Columns

- The grid has 12 columns per row.
- You define how many columns each item takes using .col-* classes.
- If you don't specify, .col automatically shares the row equally.

.col → Auto size (equal space)

.col-4 → Fixed 4/12 width

.col-sm-6 → 6/12 width starting at "sm" breakpoint

.col-md-8 → 8/12 width starting at "md" breakpoint

Breakpoints

Bootstrap uses mobile-first breakpoints (small to large).

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

Column Behavior and Features

Auto-layout

Columns without numbers (.col) automatically divide available space equally.

Variable Width

You can make columns fit their content with .col-auto.

Offsets

Offsets add left margin to shift columns horizontally.
example: <div class="col-md-4 offset-md-2">Shifted Column</div>

Column Behavior and Features

Ordering

Reorder columns visually without changing HTML structure.

example: <div class="col order-2">Second visually</div>

 <div class="col order-1">First visually</div>

Alignment

Using flexbox utilities:

- .align-items-start, .align-items-center, .align-items-end
- .justify-content-start, .justify-content-center, .justify-content-end

Spacing Utilities

Once you've placed elements with the grid, you'll notice things may be too close together or too far apart. Bootstrap solves this with margin (m-) and padding (p-) utilities.

Key concepts:
- Syntax: {property}{sides}-{size}
- Example: mt-3 = margin-top, size 3
- Sides:
    - t = top
    - b = bottom
    - l = left
    - r = right
    - x = left & right
    - y = top & bottom
    - no side = all sides
- Responsive: mt-md-5 → margin-top size 5 starting at md breakpoint.

Flexbox Utilities

The grid is powered by Flexbox, and Bootstrap exposes that directly.

Key utilities:

- Direction: flex-row, flex-column
- Horizontal alignment: justify-content-start/center/end/between/around
- Vertical alignment: align-items-start/center/end
- Ordering: order-1, order-2
- Responsive control: flex-md-column, justify-content-lg-between

Core Components Overview

- Alerts – Colored notification banners
- Badges – Small count/label indicators
- Breadcrumbs – Navigation hierarchy
- Buttons – Clickable actions (solid, outline, sizes)
- Button Groups – Grouped buttons in a line
- Cards – Content containers with images, text, actions
- Carousel – Slideshow of images/content
- Collapse – Expand/collapse content (accordion-like)
- Dropdowns – Toggle menus for actions/links
- Forms – Inputs, selects, checkboxes, textareas

Core Components Overview

- List Groups – Styled lists for menus or data
- Modal – Pop-up dialog windows
- Navs & Navbars – Horizontal/vertical navigation bars
- Pagination – Page number navigation
- Progress Bars – Visual loading bars
- Spinners – Loading indicators
- Tables – Styled tabular data
- Toasts – Small, dismissible notifications
- Tooltips & Popovers – Hover or click info bubbles