BOOTSTRAP

Grid, Components and Utilities, explained simply

Bootstrap Notes · easy to understand · free for students

I. Start with Bootstrap

- **Bootstrap** is a free CSS and JavaScript framework. It gives ready-made classes for layout, buttons, forms, menus and more.
- You write the HTML and just add class names. No need to write long CSS. Bootstrap is **mobile first** and **responsive** by default.

IDEA Think of Lego blocks. Bootstrap gives the blocks (grid, buttons, cards). You snap them together with class names to build a page fast.

Part	Job	Example
Grid	Page layout in rows and columns	`container, row, col`
Components	Ready UI pieces	`navbar, card, modal`
Utilities	Small helper classes	`mt-3, text-center`
JavaScript	Interactive parts	`dropdown, carousel`

- Add Bootstrap with a **CDN** link (easiest) or download the files. Put the CSS in `<head>` and the JS bundle before `</body>`.

EXAMPLE Starter template with the CDN (Bootstrap 5.3).

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>My Page</title>
  <link rel="stylesheet"
    href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css">
</head>
<body>
  <h1 class="text-center mt-4">Hello Bootstrap</h1>

  <script
    src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js">
  </script>
</body>
</html>
```

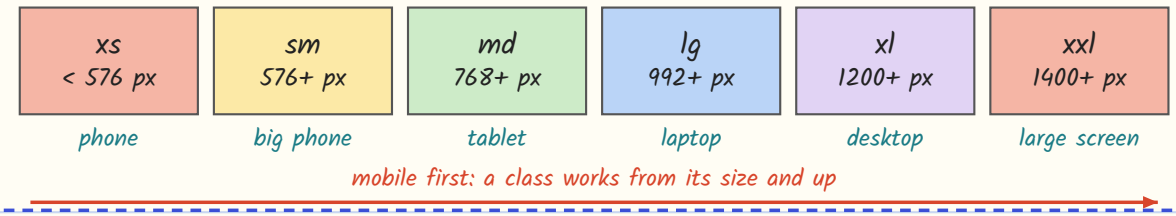
REMEMBER Always add the viewport meta tag, or the page will not look right on phones. Bootstrap 5 does not need jQuery. The `bundle` JS file already includes Popper for dropdowns and tooltips.

2. Containers and Breakpoints

- A container centers your content and gives side space. Every grid starts inside a container.

Class	Behaviour
`container`	Fixed max width that changes at each breakpoint
`container-fluid`	Always 100% width
`container-md`	100% wide until md, then fixed width (also sm, lg, xl, xxl)

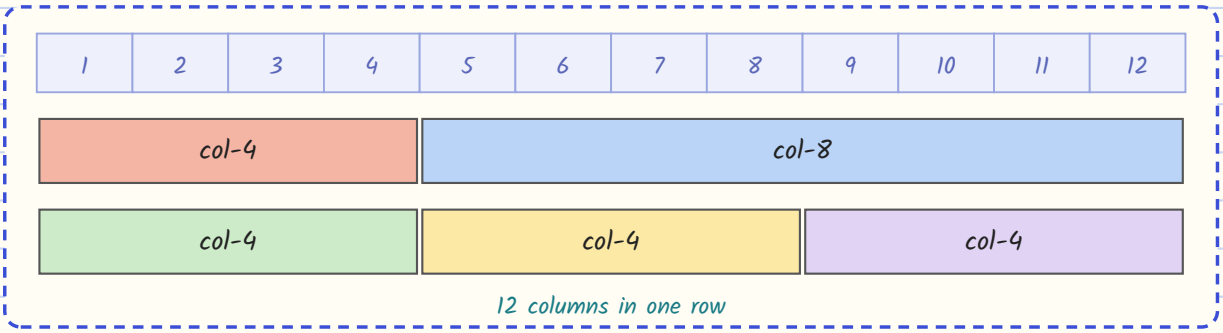
- A breakpoint is a screen width where the layout can change. Bootstrap is mobile first: a class like `col-md-6` works at md and every larger screen.



Breakpoint	Infix	Container max width
Extra small (<576px)	(none)	100%
Small (576px and up)	sm	540px
Medium (768px and up)	md	720px
Large (992px and up)	lg	960px
Extra large (1200px and up)	xl	1140px
XX large (1400px and up)	xxl	1320px

3. The Grid System

- The grid is built with flexbox. Structure is always `container` > `row` > `col`. Each row has 12 columns.
- Columns inside a row add up to 12. If they add up to more than 12, the extra columns wrap to the next line.



12 columns in one row

Class	Meaning
`col`	Equal width columns, share the row automatically
`col-4`	Takes 4 of 12 columns on all screens
`col-md-6`	6 of 12 columns from md and up. Full width below md
`col-auto`	Width of its content only
`row-cols-3`	Put 3 columns in each row without writing col classes

EXAMPLE *Equal columns and fixed columns.*

```
<div class="container">
  <div class="row">
    <div class="col">1 of 3</div>
    <div class="col">2 of 3</div>
    <div class="col">3 of 3</div>
  </div>
  <div class="row">
    <div class="col-4">4 columns</div>
    <div class="col-8">8 columns</div>
  </div>
</div>
```

➤ Responsive columns

- *Add more than one class to change the layout per screen. Read it from small to large.*

EXAMPLE *Stack on phone, 2 columns on tablet, 3 on laptop.*

```
<div class="row">
  <div class="col-12 col-md-6 col-lg-4">Box 1</div>
  <div class="col-12 col-md-6 col-lg-4">Box 2</div>
  <div class="col-12 col-md-12 col-lg-4">Box 3</div>
</div>
<!-- phone: 1 per row, md: 2 + 1, lg: 3 in one row -->
```

Gutters, offset, order and alignment

Class	What it does
g-3, gx-3, gy-3	Space (gutter) between columns: both, horizontal, vertical
offset-md-2	Push a column right by 2 columns
order-1, order-md-2	Change the visual order
align-items-center	Center columns up-down (put on row)
justify-content-between	Spread columns left to right (put on row)
w-100	Force the next columns to a new line

EXAMPLE Gutters, centering and nesting.

```
<div class="row g-3 justify-content-center align-items-center">
  <div class="col-md-4 offset-md-1">Pushed by 1</div>
  <div class="col-md-6">
    <div class="row">                        <!-- nested row inside a col -->
      <div class="col-6">Inner A</div>
      <div class="col-6">Inner B</div>
    </div>
  </div>
</div>
```

REMEMBER Put `row` inside a `container`. Put columns only inside a `row`. A nested row also gets its own 12 columns.

4. Typography, Colors and Backgrounds

- Bootstrap styles headings and text by default. Use classes to change look without writing CSS.

Class	Use
h1 to h6	Heading look on any tag, like <p class="h3">
display-1 to display-6	Very large heading text
lead	Bigger intro paragraph
fw-bold, fw-light	Font weight
fst-italic	Italic text
text-center, text-end	Text alignment (start, center, end)
text-uppercase, text-lowercase	Change letter case
text-decoration-none	Remove underline from links

Theme colors

Name	Text class	Background class
primary (blue)	`text-primary`	`bg-primary`
secondary (grey)	`text-secondary`	`bg-secondary`
success (green)	`text-success`	`bg-success`
danger (red)	`text-danger`	`bg-danger`
warning (yellow)	`text-warning`	`bg-warning`
info, light, dark	`text-info ...`	`bg-info ...`

EXAMPLE Text and background colors.

```
<h1 class="display-4 fw-bold text-primary">Big Title</h1>
<p class="lead">A larger intro line.</p>
<p class="text-danger">Error text</p>
<div class="bg-success text-white p-3">White text on green</div>
<div class="text-bg-dark p-3">text-bg sets matching text and background</div>
```

REMEMBER Use `text-bg-primary` to get a background and a readable text color together. Bootstrap 5.3 also has dark mode with `data-bs-theme="dark"`.

5. Spacing, Sizing and Display

- Spacing classes follow one pattern: property][sides]-[size. Learn it once and use it everywhere.

Part	Values	Meaning
Property	`m, p`	margin (outside), padding (inside)
Sides	`t, b, s, e, x, y`	top, bottom, start (left), end (right), x-axis, y-axis
Size	`0, 1, 2, 3, 4, 5, auto`	0, 0.25rem, 0.5rem, 1rem, 1.5rem, 3rem, auto

EXAMPLE Spacing examples.

```
<div class="mt-3">margin top 1rem</div>
<div class="p-4">padding 1.5rem on all sides</div>
<div class="px-5 py-2">padding left-right 3rem, top-bottom 0.5rem</div>
<div class="mx-auto" style="width:200px">centered with auto margins</div>
<div class="mb-md-5">bottom margin 3rem from md and up</div>
```

Sizing and display

Class	Meaning
`w-25, w-50, w-75, w-100`	Width as a percent of the parent
`h-100, vh-100`	Height of the parent, height of the screen
`mw-100`	max-width 100%
`d-none`	Hide the element
`d-block, d-inline, d-inline-block, d-flex`	Set the display type
`d-none d-md-block`	Hidden on phones, visible from md and up

EXAMPLE Show different things on different screens.

```
<p class="d-md-none">Only on phones and small screens</p>
<p class="d-none d-md-block">Only on tablets and bigger</p>
<img src="cat.jpg" class="img-fluid w-50" alt="A cat">
```

6. Flex Utilities

- Add `d-flex` to a parent and its children become flex items. These classes are the same ideas as CSS flexbox.

Class	Meaning
`flex-row, flex-column`	Direction of the main axis
`justify-content-start / center / end / between / around`	Space along the main axis
`align-items-start / center / end / stretch`	Space along the cross axis
`flex-wrap`	Let items go to the next line
`gap-3`	Space between items
`flex-fill, flex-grow-1`	Take the free space
`ms-auto, me-auto`	Push an item to the right or left

EXAMPLE Center anything on the page.

```
<div class="d-flex justify-content-center align-items-center vh-100">
  <h1>I am in the center</h1>
</div>
```

EXAMPLE *A simple top bar with a button at the end.*

```
<div class="d-flex align-items-center gap-3 p-3 bg-light">
  <strong>Logo</strong>
  <a href="#">Home</a>
  <a href="#">About</a>
  <button class="btn btn-primary ms-auto">Login</button>
</div>
```

REMEMBER *Use flex utilities for rows of items like bars and button groups. Use the grid for page columns.*

7. Borders, Shadows and Position

Class	Effect
`border, border-2, border-0`	Add border, thicker border, remove border
`border-top, border-bottom`	Border on one side
`border-primary`	Border color from the theme
`rounded, rounded-3, rounded-circle, rounded-pill`	Round the corners
`shadow-sm, shadow, shadow-lg`	Small, normal, large shadow
`position-relative / absolute / fixed / sticky`	Set the position type
`top-0, start-0, end-0, bottom-0`	Place a positioned element
`fixed-top, sticky-top`	Stick to the top of the screen

EXAMPLE Card look and a notification dot.

```
<div class="border border-primary rounded-3 shadow-sm p-4">A neat box</div>

<button class="btn btn-dark position-relative">
  Inbox
  <span class="position-absolute top-0 start-100 translate-middle badge
               rounded-pill bg-danger">5</span>
</button>
<img src="me.jpg" class="rounded-circle" width="80" alt="Me">
```

8. Buttons, Badges and Alerts

- Add `btn` plus a color class to any `<button>` or `<a>`. Outline versions have a border only.

Class	Result
`btn btn-primary`	Solid blue button (also secondary, success, danger...)
`btn btn-outline-primary`	Border only, fills on hover
`btn-lg, btn-sm`	Large or small button
`w-100 or d-grid`	Full width button
`disabled`	Greyed out and not clickable
`btn-close`	Small X close button

EXAMPLE *Buttons.*

```
<button type="button" class="btn btn-primary">Save</button>
<button type="button" class="btn btn-outline-danger btn-sm">Delete</button>
<a href="#" class="btn btn-success btn-lg disabled">Go</a>
<div class="d-grid"><button class="btn btn-dark">Full width</button></div>
```

Badges and alerts

- *A badge is a small label or count. An alert is a message box with a color for its meaning.*

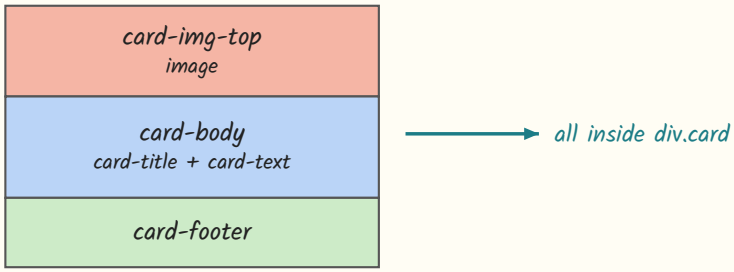
EXAMPLE *Badge and alerts.*

```
<h5>New <span class="badge text-bg-warning">12</span></h5>

<div class="alert alert-success" role="alert">Saved successfully!</div>
<div class="alert alert-danger alert-dismissible fade show" role="alert">
  Something went wrong.
  <button type="button" class="btn-close" data-bs-dismiss="alert"></button>
</div>
```

9. Cards, List Groups and Images

- *A card is a box for content. Parts:* `card`, `card-img-top`, `card-body`, `card-title`, `card-text`, `card-header`, `card-footer`.



EXAMPLE *One card.*

```
<div class="card" style="width: 18rem;">
  <img src="photo.jpg" class="card-img-top" alt="Photo">
  <div class="card-body">
    <h5 class="card-title">Card title</h5>
    <p class="card-text">Short text about this card.</p>
    <a href="#" class="btn btn-primary">Read more</a>
  </div>
</div>
```

A responsive gallery of cards

EXAMPLE *1 card on phone, 2 on tablet, 3 on laptop.*

```
<div class="row row-cols-1 row-cols-md-2 row-cols-lg-3 g-4">
  <div class="col">
    <div class="card h-100"><div class="card-body">Card 1</div></div>
  </div>
  <div class="col">
    <div class="card h-100"><div class="card-body">Card 2</div></div>
  </div>
  <div class="col">
    <div class="card h-100"><div class="card-body">Card 3</div></div>
  </div>
</div>
```

List group and images

Class	Use
`list-group, list-group-item`	Clean list of items, can be links or buttons
`active, list-group-item-action`	Highlight an item, add hover effect
`img-fluid`	Image shrinks to fit its parent (responsive)
`img-thumbnail`	Image with a border and padding
`rounded, rounded-circle`	Rounded or round image

EXAMPLE List group.

```
<ul class="list-group">
  <li class="list-group-item active">Dashboard</li>
  <li class="list-group-item">Profile</li>
  <li class="list-group-item d-flex justify-content-between">
    Messages <span class="badge text-bg-primary">3</span>
  </li>
</ul>
```

10. Navbar and Navigation

- The navbar is a menu bar. It collapses into a hamburger button on small screens. The toggler button needs `data-bs-toggle` and `data-bs-target`.

Class	Meaning
`navbar`	Starts the navbar
`navbar-expand-lg`	Full menu from lg and up, hamburger below
`navbar-brand`	Logo or site name
`navbar-toggler`	The hamburger button
`collapse navbar-collapse`	The part that hides and shows
`nav-link, active`	A menu link, current page link

EXAMPLE *Responsive navbar.*

```
<nav class="navbar navbar-expand-lg bg-dark" data-bs-theme="dark">
  <div class="container">
    <a class="navbar-brand" href="#">MySite</a>
    <button class="navbar-toggler" type="button"
            data-bs-toggle="collapse" data-bs-target="#menu">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="menu">
      <ul class="navbar-nav ms-auto">
        <li class="nav-item"><a class="nav-link active" href="#">Home</a></li>
        <li class="nav-item"><a class="nav-link" href="#">About</a></li>
        <li class="nav-item"><a class="nav-link" href="#">Contact</a></li>
      </ul>
    </div>
  </div>
</nav>
```

Tabs, pills, breadcrumb and pagination

Component	Main classes
Tabs	`nav nav-tabs, nav-link, tab-content, tab-pane`
Pills	`nav nav-pills`
Breadcrumb	`breadcrumb, breadcrumb-item`
Pagination	`pagination, page-item, page-link`

EXAMPLE Pills and breadcrumb.

```
<ul class="nav nav-pills">
  <li class="nav-item"><a class="nav-link active" href="#">One</a></li>
  <li class="nav-item"><a class="nav-link" href="#">Two</a></li>
</ul>
<nav aria-label="breadcrumb">
  <ol class="breadcrumb">
    <li class="breadcrumb-item"><a href="#">Home</a></li>
    <li class="breadcrumb-item active">Library</li>
  </ol>
</nav>
```

II. Forms

- Style each input with *form-control*. Use *form-label* for labels, `form-select` for dropdowns and `form-check` for checkboxes and radios.

Class	Use
`form-control`	Text, email, password, textarea inputs
`form-select`	Styled dropdown
`form-check, form-check-input`	Checkbox or radio with its label
`form-text`	Small help text under a field
`input-group`	Add text or buttons next to an input
`form-floating`	Label that floats inside the input
`form-control-lg / sm`	Large or small input
`is-valid, is-invalid`	Show green or red validation state

EXAMPLE *Sign up form using the grid.*

```
<form class="row g-3">
  <div class="col-md-6">
    <label for="email" class="form-label">Email</label>
    <input type="email" class="form-control" id="email" placeholder="you@mail.com">
  </div>
  <div class="col-md-6">
    <label for="pass" class="form-label">Password</label>
    <input type="password" class="form-control" id="pass">
  </div>
  <div class="col-md-8">
    <label for="city" class="form-label">City</label>
    <select id="city" class="form-select"><option>Delhi</option><option>Pune</option></select>
  </div>
  <div class="col-12">
    <div class="form-check">
      <input class="form-check-input" type="checkbox" id="ok">
      <label class="form-check-label" for="ok">I agree</label>
    </div>
  </div>
  <div class="col-12"><button class="btn btn-primary">Sign up</button></div>
</form>
```

EXAMPLE *Input group and floating label.*

```
<div class="input-group mb-3">
  <span class="input-group-text">@</span>
  <input type="text" class="form-control" placeholder="Username">
</div>
<div class="form-floating">
  <input type="email" class="form-control" id="fl" placeholder="Email">
  <label for="fl">Email address</label>
</div>
```

REMEMBER *The `for` of a label must match the `id` of its input. In a form with grid columns, put `row g-3` on the form.*

12. Tables

Class	Effect
`table`	Base Bootstrap table style (always add this)
`table-striped`	Alternate row colors
`table-hover`	Highlight a row on hover
`table-bordered`	Borders on all cells
`table-dark, table-primary`	Dark or colored table, row or cell
`table-sm`	Compact table
`table-responsive`	Scrolls sideways on small screens (put on a wrapper div)

EXAMPLE A responsive striped table.

```
<div class="table-responsive">
  <table class="table table-striped table-hover">
    <thead class="table-dark">
      <tr><th>#</th><th>Name</th><th>City</th></tr>
    </thead>
    <tbody>
      <tr><td>1</td><td>Asha</td><td>Delhi</td></tr>
      <tr><td>2</td><td>Ravi</td><td>Pune</td></tr>
    </tbody>
  </table>
</div>
```

13. JavaScript Components

- Bootstrap's JS works with data attributes. You write no JavaScript. Make sure you added the `bootstrap.bundle.min.js` file.

Attribute	Meaning
`data-bs-toggle`	What to do: modal, collapse, dropdown, tab, tooltip
`data-bs-target`	Which element to control, like #myModal
`data-bs-dismiss`	Close the modal or alert
`data-bs-parent`	Keep only one accordion item open

Modal

EXAMPLE *A popup dialog opened by a button.*

```
<button class="btn btn-primary" data-bs-toggle="modal" data-bs-target="#demo">
  Open
</button>

<div class="modal fade" id="demo" tabindex="-1">
  <div class="modal-dialog modal-dialog-centered">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title">Hello</h5>
        <button class="btn-close" data-bs-dismiss="modal"></button>
      </div>
      <div class="modal-body">This is a modal.</div>
      <div class="modal-footer">
        <button class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
      </div>
    </div>
  </div>
</div>
```

Dropdown, collapse and accordion

EXAMPLE *Dropdown menu and a collapsible box.*

```
<div class="dropdown">
  <button class="btn btn-secondary dropdown-toggle" data-bs-toggle="dropdown">
    Menu
  </button>
  <ul class="dropdown-menu">
    <li><a class="dropdown-item" href="#">Action</a></li>
    <li><a class="dropdown-item" href="#">Another</a></li>
  </ul>
</div>

<button class="btn btn-link" data-bs-toggle="collapse" data-bs-target="#more">
  Show more
</button>
<div class="collapse" id="more"><div class="card card-body">Hidden text</div></div>
```

EXAMPLE *Accordion with one item open at a time.*

```
<div class="accordion" id="acc">
  <div class="accordion-item">
    <h2 class="accordion-header">
      <button class="accordion-button" data-bs-toggle="collapse" data-bs-target="#a1">
        Question 1
      </button>
    </h2>
    <div id="a1" class="accordion-collapse collapse show" data-bs-parent="#acc">
      <div class="accordion-body">Answer 1</div>
    </div>
  </div>
</div>
```

Carousel and tooltip

- A carousel is an image slider (`carousel slide`, `carousel-item`, `carousel-control-prev`). A tooltip is a small hint on hover and needs one line of JavaScript to start.

EXAMPLE Tooltip: add the attribute, then start it.

```
<button class="btn btn-dark" data-bs-toggle="tooltip" title="I am a tooltip">
  Hover me
</button>

<script>
  document.querySelectorAll('[data-bs-toggle="tooltip"]')
    .forEach(el => new bootstrap.Tooltip(el));
</script>
```

REMEMBER Modal, dropdown, collapse and tabs work with just data attributes. Tooltips and popovers need JavaScript to start.

14. Projects and Practice

Project	What you practice
1. Personal portfolio page	Container, grid, typography, images
2. Responsive navbar site	Navbar, collapse, spacing utilities
3. Landing page	Grid, buttons, cards, display utilities
4. Pricing cards	Cards, flex, shadows, badges
5. Contact or sign up form	Forms, grid, validation
6. Admin dashboard	Table, list group, modal, dropdown
7. Photo gallery with slider	Row cols, img-fluid, carousel

Common mistakes

Mistake	Fix
No viewport meta tag	Add it in the head
Using col outside a row	Always container > row > col
Columns add up to more than 12	They wrap. Check your numbers
Using ml-3 and mr-3 (Bootstrap 4)	Bootstrap 5 uses ms-3 and me-3
Loading JS before the page needs it	Put the bundle script before </body>
Plain css hard to override	Load your style.css after Bootstrap

Practice questions

1. What is Bootstrap? Why is it called mobile first?
2. Explain the 12 column grid with container, row and col.
3. What are the Bootstrap breakpoints? What does col-md-6 mean?
4. What is the difference between container and container-fluid?
5. Explain the spacing class pattern. What does px-3 mt-5 do?
6. How do you center a box on the page using flex utilities?
7. Build a navbar that collapses on mobile.
8. Build a card gallery: 1 card on phone, 3 on laptop.
9. How does a modal open without writing JavaScript?
10. What changed from Bootstrap 4 to 5 (jQuery, ml/mr, data-bs)?

- Open the Bootstrap docs next to your code editor. Build one small page every week and copy a real website's layout for practice.