

# 1. Basic Elements of the Bootstrap Grid System

## Container

A **container** is a wrapper that holds all your page content. There are two types of containers:

- `.container` : A fixed-width container that is centered according to the screen size.
- `.container-fluid` : A full-width container that spans the entire width of the viewport.

## Row

A **row** is the logical section where you place your columns. You create a row using the `.row` class. The columns inside a row are divided into a 12-column grid model.

## Column

Columns are the core components of the Grid System. Each row is divided into a total of 12 columns. You can partition these columns into various widths to display your content in different sizes. Different column classes are used for different screen sizes.

---

# 2. Mobile Devices and Responsive Classes

Bootstrap is a **mobile-first** framework. This means it is designed primarily for mobile (small screens) and then styles are progressively applied for larger screens.

## Classes for Mobile Devices

In Bootstrap 5, there is no separate class for extra small (xs) devices (such as `.col-xs-`). Instead, you use the default `.col-` class. For example:

- `.col-12` : On mobile or extra small screens, the column will span the full 12-grid width.
- `.col-6` : On mobile devices, the column will span 6 grid columns, which is half the width.

For larger screens, you can use breakpoint-specific classes:

- `.col-sm-` : For small devices ( $\geq 576\text{px}$ )
- `.col-md-` : For medium devices ( $\geq 768\text{px}$ )
- `.col-lg-` : For large devices ( $\geq 992\text{px}$ )
- `.col-xl-` : For extra large devices ( $\geq 1200\text{px}$ )
- `.col-xxl-` : For extra extra large devices ( $\geq 1400\text{px}$ )

## Example:

Below is an example that demonstrates how the column widths change from mobile devices to larger screens:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Bootstrap Grid Example</title>
  <!-- Bootstrap CSS CDN -->
  <link href="
<https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css>"
rel="stylesheet">
</head>
<body>
  <div class="container">
    <div class="row">
      <!-- Extra Small (Mobile): col-12, Small (Tablet): col-sm-6, Medium
(Near Desktop): col-md-4 -->
      <div class="col-12 col-sm-6 col-md-4">
        Column 1
      </div>
      <div class="col-12 col-sm-6 col-md-4">
        Column 2
      </div>
      <div class="col-12 col-sm-12 col-md-4">
        Column 3
      </div>
    </div>
  </div>

  <!-- Bootstrap JS (optional) -->
  <script src="
<https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js>"></script>
</body>
</html>
```

### Explanation:

- **Mobile (Extra Small):** Since each column has the class `.col-12`, on mobile or very small screens, each column will take up the entire width (12 out of 12) and will be stacked vertically.
- **Tablet (Small, ≥576px):** The first two columns use `.col-sm-6`, so on tablets each will take up half the width (6 out of 12). The third column uses `.col-sm-12`, so it takes the full width (12 out of 12).

- **Desktop or Larger Devices (Medium, ≥768px):** All three columns use `.col-md-4`, meaning each column occupies 4 grid columns (4+4+4=12) on medium and larger screens.
- 

## 3. Steps to Use the Grid System

### 1. Include Bootstrap CSS

Add the Bootstrap CSS link in the `<head>` section of your HTML document.

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

### 2. Create a Container

Wrap your content within a container.

```
<div class="container">
  <!-- Your content goes here -->
</div>
```

### 3. Create a Row

Inside the container, create a row.

```
<div class="row">
  <!-- Place your columns here -->
</div>
```

### 4. Add Columns

Inside the row, use the appropriate column classes for mobile to larger devices.

```
<div class="row">
  <div class="col-12 col-sm-6">
    On mobile: 12 (full width), on small screens: 6 (half width)
  </div>
  <div class="col-12 col-sm-6">
    On mobile: 12 (full width), on small screens: 6 (half width)
  </div>
</div>
```

---

## 4. Key Points

- **Mobile-first Approach:** Bootstrap is designed by default for mobile (extra small) screens, and then additional classes are added to customize layouts for larger screens.
- **12-Column Grid:** Each row is divided into a maximum of 12 columns. You can split these 12 columns in various ways to design different layouts.
- **Responsive Design:** Use breakpoint-specific classes (such as `.col-sm-`, `.col-md-`, etc.) to make your layout responsive so that it looks great on mobile devices, tablets, and desktops.