



BAHIR DAR UNIVERSITY

Bahir Dar Institute of Technology (BiT)

Faculty of Computing

Department of Software Engineering

Web Design and Programming

Bootstrap Documentation

Documented by: -

1. Abrham Desalegn.....1307050
2. Abiy Shiferaw.....1306078
3. Jemal Workie.....1307712
4. Smachew Gedefaw.....1308736
5. Solomon Muhye.....1309375
6. Yordanos Ayenew.....1311662

Submitted to: Instructor Tesfahunegn

Submit date: January 23, 2023

Contents

Introduction	2
Requirements to use bootstrap	3
Installation	4
Bootstrap syntaxes	6
1. Containers	6
2. Grid system	6
3. Typography	8
4. Colors	9
a. Text colors	9
b. Background colors	9
5. Buttons	10
6. Forms	11
7. Images and media	11
8. Navbar	12
9. Modal	14
10. Jumbotron	16
11. Flexbox	16
12. Utilities	19
Summary	23
References	24

Introduction

Bootstrap is a framework that is used to create websites and web applications. It is a collection of tools and pre-designed components that make it easy to create and design websites that look good and work well on a variety of devices. Bootstrap was first developed by Twitter and it has since become one of the most popular front-end web development frameworks in use today.

Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins.

One of the key features of Bootstrap is its responsive grid system. The grid system allows developers to create layouts that automatically adapt to different screen sizes. This means that a website created with Bootstrap will look good and work well on large and small screens, such as desktop computers, laptops, tablets, and smartphones. The grid system is based on a 12-column layout, and it includes a variety of classes that can be used to create different types of layouts. This allows developers to easily create flexible and responsive designs without having to write a lot of complex CSS code.

Another important feature of Bootstrap is its pre-designed UI components. These components include forms, buttons, navigation menus, and models, and they are designed to be easy to use and customizable. They are also designed to be responsive and mobile-friendly, so they will work well on small screens. This means that developers can add functionality and style to their websites with minimal effort.

Bootstrap also includes a number of JavaScript plugins that can be used to add additional functionality to a website, such as modals, tooltips, and carousels. These plugins are designed to be easy to use and can be easily integrated into a website using the Bootstrap JavaScript library. This means that developers can add interactive and dynamic features to their websites without needing to write a lot of complex JavaScript code.

Bootstrap is widely used and is often considered one of the most popular front-end web development frameworks. It is popular because it is easy to learn and use, it can save a lot of time and effort when building websites, and it is supported by a large and active community of developers. Bootstrap 5 is the latest version of Bootstrap that offers improved accessibility, new and better customization options, and many other new features and improvements.

Bootstrap is a powerful and flexible framework that makes creating responsive, mobile-friendly websites easy. It provides a set of tools and pre-designed components that make it easy to create and design websites that look good and work well on a variety of devices. With Bootstrap, you can create a website that looks good and works well on a wide range of devices, without needing to write a lot of complex CSS or JavaScript code. It's a great tool for both beginners and experienced developers, it saves time and makes it easy to create professional and responsive designs.

Responsive web design is about creating web sites which automatically adjust themselves to look good on all devices, from small phones to large desktops.

Advantages of Bootstrap

- **Easy to use:** Anybody with just basic knowledge of HTML and CSS can start using Bootstrap
- **Responsive features:** Bootstrap's responsive CSS adjusts to phones, tablets, and desktops
- **Mobile-first approach:** In Bootstrap, mobile-first styles are part of the core framework
- **Browser compatibility:** Bootstrap 5 is compatible with all modern browsers (Chrome, Firefox, Edge, Safari, and Opera). **Note** that if you need support for IE11 and down, you must use either BS4 or BS3.

Requirements to use bootstrap

In detail, to use Bootstrap you will need the following:

1. A web development environment: This can include an HTML editor, CSS editor, and JavaScript editor. You can use a standalone editor for each, or use an integrated development environment (IDE) that includes all three. Some popular options include Visual Studio Code, Atom, and Sublime Text.
2. A web browser: Bootstrap is a front-end framework, so you will need a web browser to view the pages you create. Bootstrap is compatible with all modern web browsers, including Chrome, Firefox, Safari, and Edge.
3. Bootstrap files: You can download the Bootstrap files from the official website and include them in your project. The files include CSS and JavaScript files that you will need to reference in your HTML.
4. Basic understanding of HTML, CSS, and JavaScript: Bootstrap is built on top of these technologies, so a basic understanding of them is required to use Bootstrap effectively.

5. Optional: A Content Delivery Network (CDN) to include the necessary files in your HTML, this is a way to host the files of Bootstrap in a server closer to your client, this is useful for reduce the time to load the page and for not overload your server.
6. Optional: A package manager like npm or yarn to easily manage and update the dependencies.

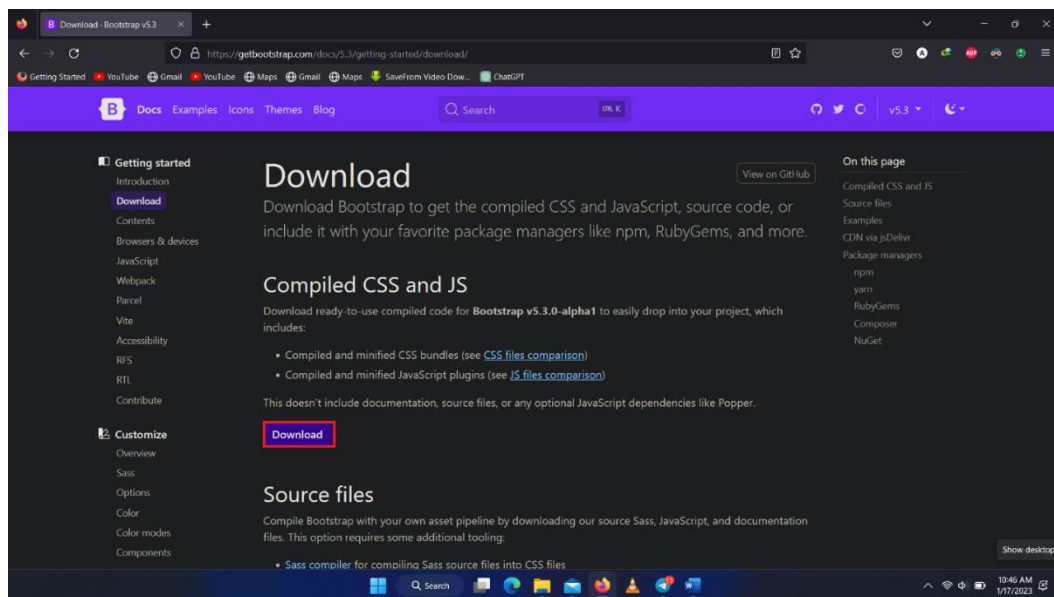
Once you have these requirements in place, you can start using Bootstrap to create responsive, mobile-friendly websites and web applications.

Installation

To install Bootstrap, you will need to include the Bootstrap CSS and JavaScript files in your HTML file.

Here are the basic steps to install Bootstrap:

1. Download the latest version of Bootstrap from the official website:
<https://getbootstrap.com/docs/5.3/getting-started/download/>



2. Extract the downloaded file and copy the CSS and JS folders.
3. In your HTML file, include the following code in the head section to link the CSS file:

```
<link rel="stylesheet" href="path/to/bootstrap.min.css">
```

4. Include the following code before the closing body tag to link the JS file:

```
<script src="path/to/jquery.min.js"></script>
<script src="path/to/bootstrap.min.js"></script>
```

Now you can start using Bootstrap classes and components in your HTML file.

Alternatively, you can use Bootstrap CDN to include the Bootstrap CSS and JS in your HTML file.

A Content Delivery Network (CDN) is a network of servers that help deliver content faster by serving it from a location closer to the user. The purpose of a CDN is to deliver content, such as web pages, images, videos, and other types of files, to users more quickly and efficiently by serving the content from a server that is geographically closer to the user.

```
<!-- Latest compiled and minified CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet">

<!-- Latest compiled JavaScript -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js"></script>
```

We can also use a package manager ² like npm or yarn to install bootstrap in our project

```
npm install bootstrap
```

or

```
yarn add bootstrap
```

A package manager is a tool that helps you manage the software libraries and dependencies that your project needs in order to function. It allows you to easily install, update, and remove libraries and dependencies with a single command, without having to manually download and manage the files yourself. Package managers also help to ensure that all the dependencies of your project are compatible and work well together.

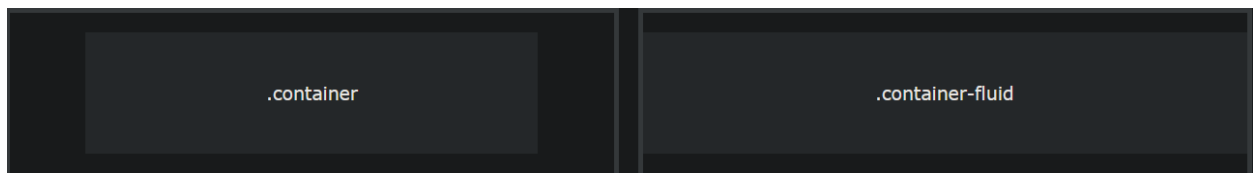
Bootstrap syntaxes

1. Containers

A container in Bootstrap is a box that helps to organize and structure the layout of a webpage. It's used to create a container for the grid system and it sets a maximum width for the grid, so it doesn't take up the whole screen, and it centers the grid horizontally on the screen.

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

1. The `.container` class provides a responsive **fixed width container**
2. The `.container-fluid` class provides a **full width container**, spanning the entire width of the viewport



By default, containers have left and right padding, with no top or bottom padding.

We can also use the `.container-sm|md|lg|xl` classes to determine when the container should be responsive. The max-width of the container will change on different screen sizes/viewports.

Class	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra large ≥1200px	XXL ≥1400px
<code>.container-sm</code>	100%	540px	720px	960px	1140px	1320px
<code>.container-md</code>	100%	100%	720px	960px	1140px	1320px
<code>.container-lg</code>	100%	100%	100%	960px	1140px	1320px
<code>.container-xl</code>	100%	100%	100%	100%	1140px	1320px
<code>.container-xxl</code>	100%	100%	100%	100%	100%	1320px

2. Grid system

The grid system in Bootstrap is a layout system that uses a 12-column grid to create responsive and flexible layouts for web pages. It allows developers to create rows and columns, and to place content within those columns.

span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1
span 4				span 4				span 4			
span 4				span 8							
span 6						span 6					
span 12											

The grid is based on a 12-column layout, where the columns are divided into 12 equal parts. This means that you can use any combination of the 12 columns to create different layouts for different screen sizes. Bootstrap also uses a responsive design, which means that the layout automatically adjusts to fit different screen sizes.

The Bootstrap 5 grid system has six classes:

- `.col`- (extra small devices - screen width less than 576px)
- `.col-sm`- (small devices - screen width equal to or greater than 576px)
- `.col-md`- (medium devices - screen width equal to or greater than 768px)
- `.col-lg`- (large devices - screen width equal to or greater than 992px)
- `.col-xl`- (xlarge devices - screen width equal to or greater than 1200px)
- `.col-xxl`- (xxlarge devices - screen width equal to or greater than 1400px)

The classes above can be combined to create more dynamic and flexible layouts. For example, you could use `.col-sm-6` to create a layout that has two columns on small screens and `.col-lg-4` to create a layout that has three columns on large screens.

Each class scales up, so if we want to set the same widths for `sm` and `md`, we only need to specify `sm`.

The columns can be managed using the `.row` class.

The `.row` class is a class that is used to create a row within the grid system. It is used to divide the grid into horizontal sections, where you can place content in columns.

When creating a grid layout using Bootstrap, we first create a container element and then add rows within that container. Each row can contain one or more columns, where we can place your content. The `.row` class is used to create a new row within the grid, and it is typically added to a `.div` element.


```

<div class="container-fluid">
  <div class="row">
    <div class="col-sm-3 p-3 bg-primary text-white">.col</div>
    <div class="col-sm-3 p-3 bg-dark text-white">.col</div>
    <div class="col-sm-3 p-3 bg-primary text-white">.col</div>
    <div class="col-sm-3 p-3 bg-dark text-white">.col</div>
  </div>
</div>

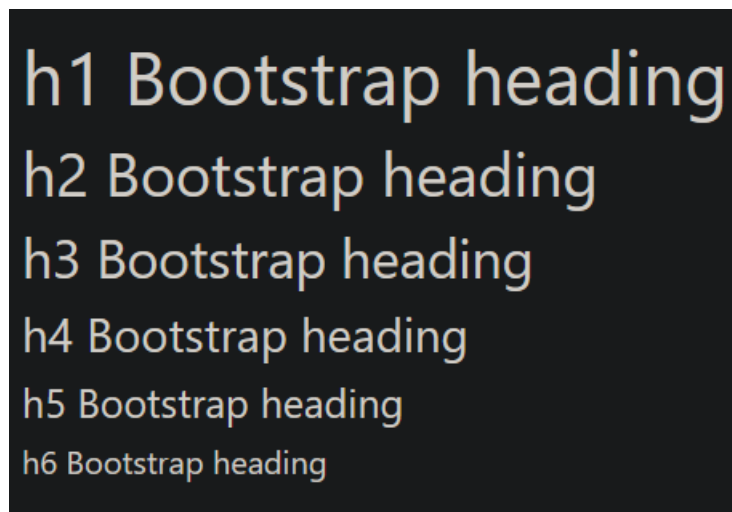
```



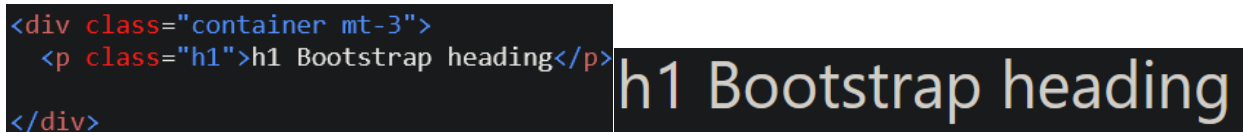
3. Typography

Typography in Bootstrap refers to the design and styling of text on a webpage. Bootstrap provides a set of pre-defined styles for headings, paragraphs, lists, and other text elements, that we can use to create a consistent and attractive layout for your webpage.

Bootstrap 5 styles HTML headings (<h1> to <h6>) with a bolder font-weight and a responsive font-size.



We can also use .h1 to .h6 classes on other elements to make them behave as headings if you want:



Other HTML elements like p, ul and ol are supported by the Bootstrap framework.

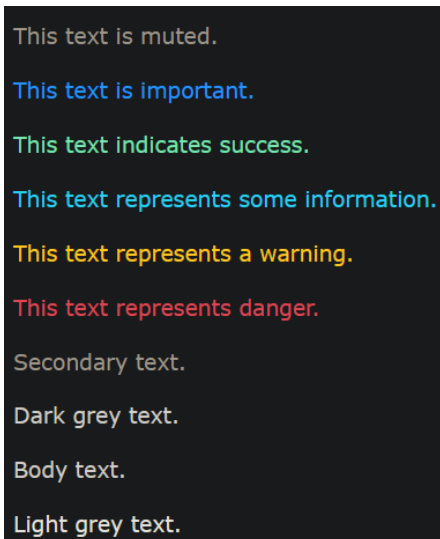
4. Colors

a. Text colors

Bootstrap 5 has some contextual classes that can be used to provide "meaning through colors".

The classes for text colors are:

- ✓ `.text-muted`
- ✓ `.text-primary`
- ✓ `.text-success`
- ✓ `.text-info`
- ✓ `.text-warning`
- ✓ `.text-danger`
- ✓ `.text-secondary`
- ✓ `.text-white`
- ✓ `.text-dark`
- ✓ `.text-body` (default body color/often black)
- ✓ `.text-light`



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.

Dark grey text.

Body text.

Light grey text.

b. Background colors

The classes for background colors are:

- ✓ `.bg-primary`
- ✓ `.bg-success`
- ✓ `.bg-info`
- ✓ `.bg-warning`
- ✓ `.bg-danger`

- ✓ `.bg-secondary`
- ✓ `.bg-dark`
- ✓ `.bg-light`



The `.bg-color` classes above does not work well with text, or at least then we have to specify a proper `.text-color` class to get the right text color for each background.

However, you can use the `.text-bg-color` classes and Bootstrap will automatically handle the appropriate text color for each background color.

5. Buttons

Bootstrap 5 provides different styles of buttons. The basic button can be styled by `.btn` class, while we can style the color of the button using the colors mentioned above.

- ✓ `.btn`
- ✓ `.btn-primary`
- ✓ `.btn-secondary`
- ✓ `.btn-success`
- ✓ `.btn-info`
- ✓ `.btn-warning`
- ✓ `.btn-danger`
- ✓ `.btn-dark`
- ✓ `.btn-light`



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

```
<a href="#" class="btn btn-success">Link Button</a>
<button type="button" class="btn btn-success">Button</button>
<input type="button" class="btn btn-success" value="Input Button">
```

We can use the `.btn-lg` class for large buttons or `.btn-sm` class for small buttons.

```
<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>
```

Large

Default

Small

6. Forms

We use `.form-control` to apply styles to form controls, such as text input fields, select boxes, and text areas, and we use `.form-label` to style the labels of form elements.

```
<label for="email" class="form-label">Email:</label>
<input type="email" class="form-control" id="email" placeholder="Enter email" name="email">
```

Email:

Enter email

`.form-check` is used to style checkboxes and radio buttons.

`.form-check-input` is used to apply styles to the checkbox or radio button input element

`.form-check-label` is used to apply styles to the checkbox or radio button label

```
<div class="form-check mb-3">
  <label class="form-check-label">
    <input class="form-check-input" type="checkbox" name="remember"> Remember me
  </label>
</div>
```

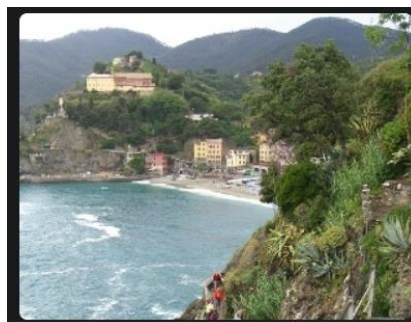
☐ Remember me

7. Images and media

The `.rounded` class adds rounded corners to an image.

```

```



The `.rounded-circle` class shapes the image to a circle

```

```



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

```

```



We can create responsive images by adding an `.img-fluid` class to the `` tag. The image will then scale nicely to the parent element.

8. Navbar

With Bootstrap, a navigation bar can extend or collapse, depending on the screen size.

A standard navigation bar is created with the `.navbar` class, followed by a responsive collapsing class: `.navbar-expand-xxl|xl|lg|md|sm` (stacks the navbar vertically on xxlarge, extra large, large, medium or small screens).

To add links inside the navbar, use either an `` element (or a `<div>`) with `class="navbar-nav"`. Then add `` elements with a `.nav-item` class followed by an `<a>` element with a `.nav-link` class.

```
<!-- A grey horizontal navbar that becomes vertical on small screens -->
<nav class="navbar navbar-expand-sm bg-light">

  <div class="container-fluid">
    <!-- Links -->
    <ul class="navbar-nav">
      <li class="nav-item">
        <a class="nav-link" href="#">Link 1</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Link 2</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Link 3</a>
      </li>
    </ul>
  </div>

</nav>
```

Link 1 Link 2 Link 3

The `.navbar-brand` class is used to highlight the brand/logo/project name of your page.

```
<nav class="navbar navbar-expand-sm bg-dark navbar-dark">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Logo</a>
  </div>
</nav>
```

Logo

Very often, especially on small screens, you want to hide the navigation links and replace them with a button that should reveal them when clicked on.

To create a collapsible navigation bar, use a button with `class="navbar-toggler"`, `data-bs-toggle="collapse"` and `data-bs-target="#thetarget"`. Then wrap the navbar content (links, etc) inside a `<div>` element with `class="collapse navbar-collapse"`, followed by an id that matches the `data-bs-target` of the button: `"thetarget"`.

```
<button
  class="navbar-toggler"
  type="button"
  data-bs-toggle="collapse"
  data-bs-target="#navmenu"
>
  <span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navmenu">
  <ul class="navbar-nav ms-auto">
    <li class="nav-item">
      <a href="#learn" class="nav-link">What You'll Learn</a>
    </li>
    <li class="nav-item">
      <a href="#questions" class="nav-link">Questions</a>
    </li>
    <li class="nav-item">
      <a href="#Instructors" class="nav-link">Instructors</a>
    </li>
  </ul>
</div>
</div>
```

9. Modal

The Modal component is a dialog box/popup window that is displayed on top of the current page.

Here are some of the most commonly used modal classes in Bootstrap:

- `.modal`: This class is used to create the overall container for the modal dialog.
- `.modal-dialog`: This class is used to create the container for the modal's content.
- `.modal-content`: This class is used to create the container for the modal's body and footer.
- `.modal-header`: This class is used to create the container for the modal's header.

- `.modal-title`: This class is used to style the title of the modal.
- `.modal-body`: This class is used to create the container for the modal's body.
- `.modal-footer`: This class is used to create the container for the modal's footer.

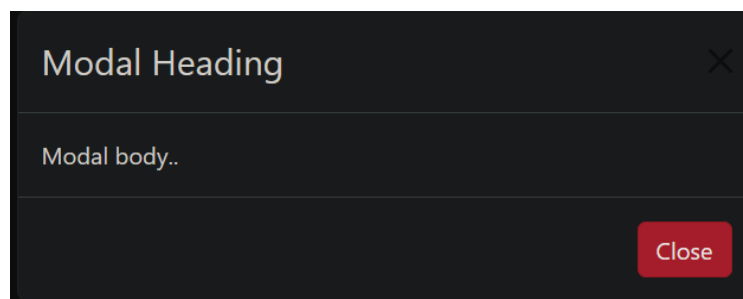
```
<!-- The Modal -->
<div class="modal" id="myModal">
  <div class="modal-dialog">
    <div class="modal-content">

      <!-- Modal Header -->
      <div class="modal-header">
        <h4 class="modal-title">Modal Heading</h4>
        <button type="button" class="btn-close" data-bs-dismiss="modal"></button>
      </div>

      <!-- Modal body -->
      <div class="modal-body">
        Modal body..
      </div>

      <!-- Modal footer -->
      <div class="modal-footer">
        <button type="button" class="btn btn-danger" data-bs-dismiss="modal">Close</button>
      </div>

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



We can use the `.fade` class to add a fading effect when opening and closing the modal.

10. Jumbotron

Jumbotron is typically used to create a full-width container that stands out from the rest of the page, and is often used to display key content such as headlines, calls-to-action, or promotional messages.

Jumbotrons are no longer supported in Bootstrap 5. **However**, we can use a `<div>` element and add special helper classes together with a color class to achieve the same effect

```
<div class="mt-4 p-5 bg-primary text-white rounded">  
  <h1>Jumbotron Example</h1>  
  <p>This is the jumbotron example</p>  
</div>
```



11. Flexbox

Flexbox in Bootstrap is a way to arrange elements within a container in a flexible and adaptable way. It allows you to align and distribute elements horizontally or vertically, and to change their order based on the size of the screen or device. It makes it easier to design flexible responsive layout structure without using float or positioning.

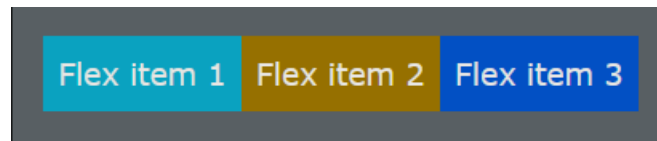
We can use the `d-flex` class to create a flexbox container and to transform direct children into flex items.

```
<div class="d-flex p-3 bg-secondary text-white">  
  <div class="p-2 bg-info">Flex item 1</div>  
  <div class="p-2 bg-warning">Flex item 2</div>  
  <div class="p-2 bg-primary">Flex item 3</div>  
</div>
```



We can use the `d-inline-flex` class to create an inline flexbox container.

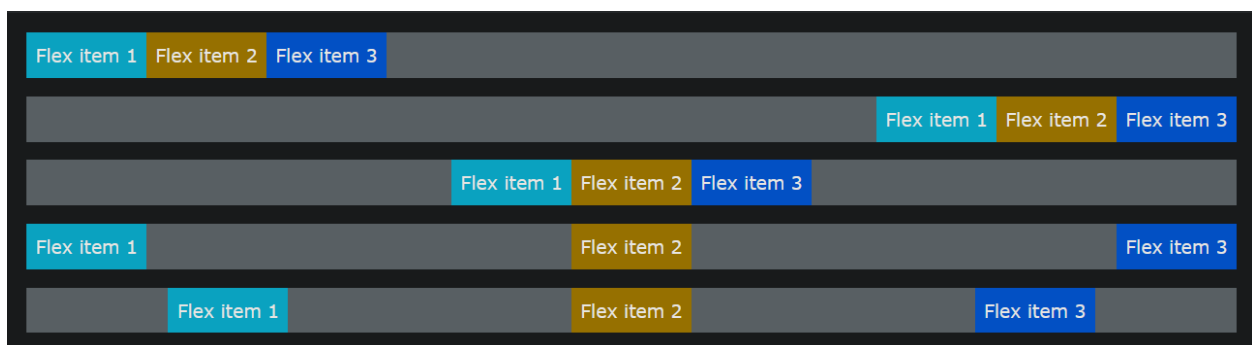
```
<div class="d-inline-flex p-3 bg-secondary text-white">
  <div class="p-2 bg-info">Flex item 1</div>
  <div class="p-2 bg-warning">Flex item 2</div>
  <div class="p-2 bg-primary">Flex item 3</div>
</div>
```



We can use the `.justify-content-*` classes to change the alignment of flex items. The valid classes are

- ✓ `.justify-content-start`
- ✓ `.justify-content-end`
- ✓ `.justify-content-center`
- ✓ `.justify-content-between`
- ✓ `.justify-content-around`

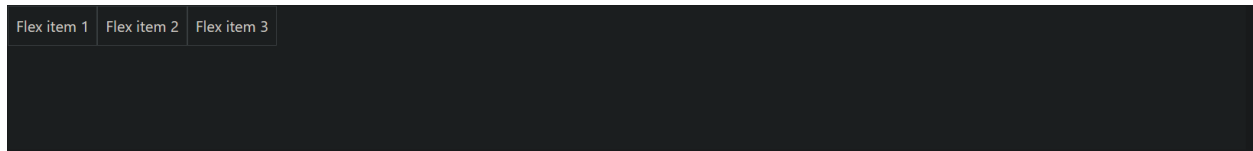
```
<div class="d-flex justify-content-start">...</div>
<div class="d-flex justify-content-end">...</div>
<div class="d-flex justify-content-center">...</div>
<div class="d-flex justify-content-between">...</div>
<div class="d-flex justify-content-around">...</div>
```



We can control the vertical alignment of **single rows** of flex items with the `.align-items-*` classes. The valid classes are

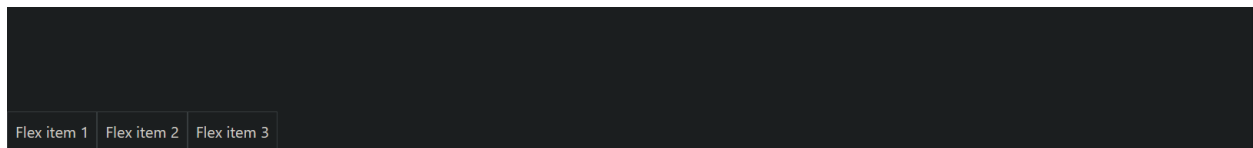
- ✓ `.align-items-start`

```
<div class="d-flex align-items-start bg-light" style="height:150px">
  <div class="p-2 border">Flex item 1</div>
  <div class="p-2 border">Flex item 2</div>
  <div class="p-2 border">Flex item 3</div>
</div>
```



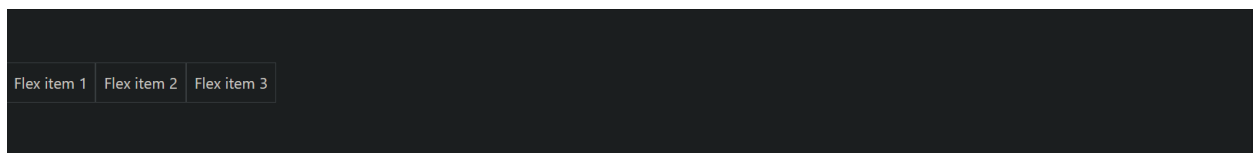
✓ .align-items-end

```
<div class="d-flex align-items-end bg-light" style="height:150px">
  <div class="p-2 border">Flex item 1</div>
  <div class="p-2 border">Flex item 2</div>
  <div class="p-2 border">Flex item 3</div>
</div>
```



✓ .align-items-center

```
<div class="d-flex align-items-center bg-light" style="height:150px">
  <div class="p-2 border">Flex item 1</div>
  <div class="p-2 border">Flex item 2</div>
  <div class="p-2 border">Flex item 3</div>
</div>
```



✓ .align-items-stretch

```
<div class="d-flex align-items-stretch bg-light" style="height:150px">
  <div class="p-2 border">Flex item 1</div>
  <div class="p-2 border">Flex item 2</div>
  <div class="p-2 border">Flex item 3</div>
</div>
```

Flex item 1	Flex item 2	Flex item 3	
-------------	-------------	-------------	--

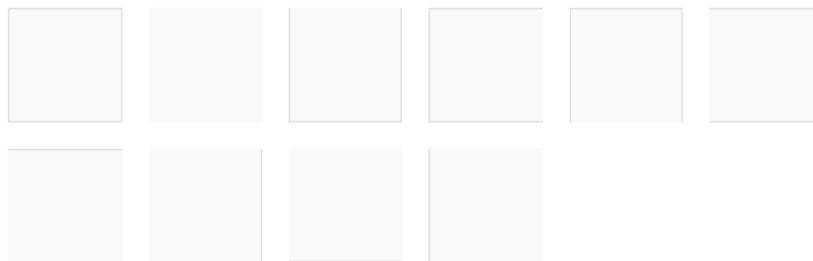
12. Utilities

Bootstrap 5 has a lot of utility/helper classes to quickly style elements without using any CSS code.

We use the `border` classes to add or remove borders from an element.

```
<span class="border"></span>
<span class="border border-0"></span>
<span class="border border-top-0"></span>
<span class="border border-end-0"></span>
<span class="border border-bottom-0"></span>
<span class="border border-start-0"></span>
<br>

<span class="border-top"></span>
<span class="border-end"></span>
<span class="border-bottom"></span>
<span class="border-start"></span>
```



We can use `.border-1` to `.border-5` to change the width of the border.

```

<span class="border border-1"></span>
<span class="border border-2"></span>
<span class="border border-3"></span>
<span class="border border-4"></span>
<span class="border border-5"></span>

```



We can also add rounded corners to an element with the `rounded` classes.

```

<span class="rounded"></span>
<span class="rounded-top"></span>
<span class="rounded-end"></span>
<span class="rounded-bottom"></span>
<span class="rounded-start"></span>
<span class="rounded-circle"></span>
<span class="rounded-pill" style="width:130px"></span>
<span class="rounded-0"></span>
<span class="rounded-1"></span>
<span class="rounded-2"></span>
<span class="rounded-3"></span>
<span class="rounded-4"></span>
<span class="rounded-5"></span>

```



We can center an element with the `.mx-auto` class (adds `margin-left: auto` and `margin-right: auto`)

```

<div class="mx-auto bg-warning" style="width:150px">Centered</div>

```

Centered

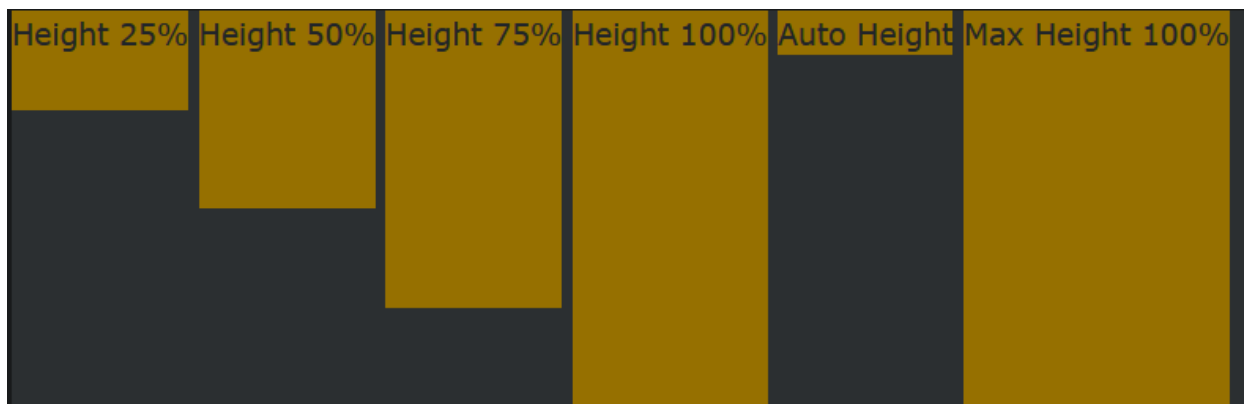
We can set the width of an element with the w-* classes (.w-25, .w-50, .w-75, .w-100, .mw-auto, .mw-100)

```
<div class="w-25 bg-warning">Width 25%</div>
<div class="w-50 bg-warning">Width 50%</div>
<div class="w-75 bg-warning">Width 75%</div>
<div class="w-100 bg-warning">Width 100%</div>
<div class="w-auto bg-warning">Auto Width</div>
<div class="mw-100 bg-warning">Max Width 100%</div>
```



We can set the height of an element with the h-* classes (.h-25, .h-50, .h-75, .h-100, .mh-auto, .mh-100)

```
<div style="height:200px;background-color:#ddd">
  <div class="h-25 bg-warning">Height 25%</div>
  <div class="h-50 bg-warning">Height 50%</div>
  <div class="h-75 bg-warning">Height 75%</div>
  <div class="h-100 bg-warning">Height 100%</div>
  <div class="h-auto bg-warning">Auto Height</div>
  <div class="mh-100 bg-warning" style="height:500px">Max Height 100%</div>
</div>
```



Bootstrap 5 has a wide range of responsive margin and padding utility classes. They work for all breakpoints:

- ✓ xs (<=576px)
- ✓ sm (>=576px)
- ✓ md (>=768px)
- ✓ lg (>=992px)
- ✓ xl (>=1200px)
- ✓ xxl (>=1400px)

The classes are used in the format:

{property}{sides}-{size} for xs

and

{property}{sides}-{breakpoint}-{size} for sm, md, lg, xl and xxl.

Where *property* is one of:

- m - sets margin
- p - sets padding

Where *sides* is one of:

- t - sets margin-top or padding-top
- b - sets margin-bottom or padding-bottom
- s - sets margin-left or padding-left
- e - sets margin-right or padding-right
- x - sets both padding-left and padding-right or margin-left and margin-right
- y - sets both padding-top and padding-bottom or margin-top and margin-bottom
- blank - sets a margin or padding on all 4 sides of the element

Where *size* is one of:

- 0 - sets margin or padding to 0
- 1 - sets margin or padding to .25rem
- 2 - sets margin or padding to .5rem
- 3 - sets margin or padding to 1rem
- 4 - sets margin or padding to 1.5rem
- 5 - sets margin or padding to 3rem

- `auto` - sets margin to auto

```
<div class="pt-4 bg-warning">I only have a top padding (1.5rem)</div>
<div class="p-5 bg-success">I have a padding on all sides (3rem)</div>
<div class="m-5 pb-5 bg-info">I have a margin on all sides (3rem) and a bottom padding (3rem)</div>
```

I only have a top padding (1.5rem)

I have a padding on all sides (3rem)

I have a margin on all sides (3rem) and a bottom padding (3rem)

We can use the `.btn-close` class to style a close icon. This is often used for alerts and modals.

```
<button type="button" class="btn-close"></button>
```



Summary

Bootstrap is a powerful and widely-used open-source front-end development framework that allows developers to create responsive and mobile-first websites quickly and efficiently. Its collection of pre-designed CSS and JavaScript components, such as navigation bars, forms, buttons, and modals, make it easy to create consistent and visually appealing layouts across all browsers and devices. Additionally, the responsive grid system and pre-designed UI elements such as typography, forms and buttons, make it easy to create custom layouts without having to start from scratch. The availability of a wide range of plugins and the ease of customization makes Bootstrap a great choice for developers looking to save time and effort when building responsive and mobile-friendly websites. Overall, the flexibility, ease of use and the wide adoption of bootstrap makes it a popular choice among developers.

References

<https://www.w3schools.com/bootstrap5/>

<https://getbootstrap.com/>