Module 8) Bootstrap Basic & Advanced

• What are the advantages of Bootstrap?

Ans: Bootstrap, a popular front-end framework, offers several advantages:

➢ **Responsive Design:** Bootstrap provides a grid system and responsive design classes, making it easier to create websites that adapt and look good on various devices and screen sizes.

➢ **Consistency and Efficiency:** It offers a collection of pre-styled components and utilities that maintain consistency across the project. Developers can quickly build and design elements without reinventing the wheel.

➢ **Customizable Components:** Bootstrap allows customization. Developers can tweak the default styles, layouts, and components to suit their project's specific needs while still leveraging Bootstrap's functionality.

➢ **Cross-Browser Compatibility:** It's designed to be compatible with various browsers, reducing the need for extensive browser testing and ensuring a more consistent experience for users.

➢ **Documentation and Community:** Bootstrap has extensive documentation and a large community of developers contributing plugins, extensions, and solutions. This makes finding help, learning, and troubleshooting easier.

➢ **Support for CSS and JavaScript Components:** Bootstrap comes with a range of CSS and JavaScript components like carousels, modals, forms, navigation bars, etc., which can be easily integrated into a project.

➢ **Faster Development:** With its predefined classes and components, Bootstrap can significantly speed up the development process, enabling faster prototyping and implementation of designs.

• What is a Bootstrap Container, and how does it work?

Ans: Containers are the most basic layout element in Bootstrap and are required when using our default grid system.

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

The .container class provides a responsive fixed width container.

The .container-fluid class provides a full width container, spanning the entire width of the viewport.

- ➢ Bootstrap Containers are the most basic layout element in Bootstrap. Bootstrap Containers are very essential and basic building blocks of bootstrap that wrap a page's content. It's responsible for setting and aligning content within it according to viewport or given device. Containers are defined within the container class (.container). In other words, we can say that containers are established the width for the layout to give the content. Elements and content are added within the container.
- ➢ Containers are used for many purposes such as –
- ➢ It requires use with the default grid system.
- ➢ To establish width for the layout to give web content.
- ➢ To provide responsive fixed behavior of any web project.
- ➢ To sets the content margin dealing with the responsive behavior of your layout.
- ➢ Bootstrap has the default class or predefined class is ".container" & ".container-fluid" class for layout. Containers are used to contain, pad, and (sometimes) center the content within them. While containers can be nested, most layouts do not require a nested container.

• **What are the default Bootstrap text settings?**

Ans:

1.Text Color:

- .text-primary: Sets text color to the primary color defined in Bootstrap.
- .text-secondary: Sets text color to the secondary color defined in Bootstrap.
- .text-success: Sets text color to indicate success.

- .text-danger: Sets text color to indicate danger or errors.
- .text-warning: Sets text color to indicate a warning.
- .text-info: Sets text color to indicate information.
- .text-dark: Sets text color to a dark shade.
- .text-muted: Sets text color to a muted or light gray.
- .text-white: Sets text color to white.

2.Font Weight:

- .font-weight-bold: Makes text bold.
- .font-weight-normal: Resets text to its default normal weight.
- .font-weight-light: Sets text to a lighter weight.

3.Text Transformation:

- .text-uppercase: Transforms text to uppercase.
- .text-lowercase: Transforms text to lowercase.

.text-capitalize: Capitalizes the first letter of each word in the text.

4.Text Alignment:

- .text-left: Aligns text to the left.
- .text-center: Centers text.
- .text-right: Aligns text to the right.
- .text-justify: Justifies text.

5.Text Overflow:

- .text-truncate: Truncates text with an ellipsis (...) if it overflows its container.

6.Text Wrapping:

- .text-wrap: Allows text to wrap within its container.

These classes can be applied directly to HTML elements to adjust the appearance and behavior of text elements without the need for writing

custom CSS, providing a quick and efficient way to style text in a Bootstrap-based project.

• What do you know about the Bootstrap Grid System?

Ans:

The Bootstrap Grid System is a responsive, mobile-first grid system that allows developers to create a responsive layout easily. It's based on a 12-column grid layout that enables the creation of complex page layouts with different column combinations across various devices and screen sizes.

Key aspects of the Bootstrap Grid System:

1.12-column Layout: The grid system is divided into 12 equal columns. Developers can create layouts by combining these columns to accommodate different content arrangements.

2.Responsive Design: It's designed to be responsive, meaning the layout adapts and adjusts to fit different screen sizes, from mobile devices to desktops, by utilizing breakpoints.

3.Containers: The grid layout is encapsulated within containers (container, container-fluid) to manage the width and responsiveness of the grid.

4.Rows and Columns: Content is organized using rows and columns. Rows are horizontal containers for columns, and columns are placed inside rows to create the layout structure.

5.Column Classes: Bootstrap provides classes such as col-{breakpoint}-{number} to define the width of columns at different breakpoints (xs, sm, md, lg, xl, xxl). For example, col-md-6 would create a column that spans half the width of its container on medium-sized screens and above.

Using the Bootstrap Grid System, developers can create flexible and responsive layouts without having to write extensive custom CSS, thus facilitating the development of mobile-friendly and visually consistent designs across various devices.

• What is the difference between Bootstrap 4 and Bootstrap 5

Ans: **1. Grid System:**

- ➢ Bootstrap 4 has 5 tier (xs, sm, md, lg, xl)**.**
- ➢ Bootstrap 5 has 6 tier (xs, sm, md, lg, xl, xxl).

2. Internet Explorer:

- ➢ Bootstrap 4 supports both IE 10 and 11.
- ➢ Bootstrap 5 doesn't support IE 10 and 11.

3. Color:

- ➢ Bootstrap 4 has limited colors
- ➢ Bootstrap 5 Extra colors added with the looks, A card improved color palette. there are various shades available to choose.

4. Bootstrap Icons:

- ➢ Bootstrap 4 doesn't have its own SVG icons, we have to use font-awesome for icons.
- ➢ Bootstrap 5 have its own SVG icons

Static Site Generator:

- ➢ Bootstrap 4 uses Jekyll software.
- ➢ Bootstrap 5 uses Hugo software as it is fast static site generator.

5.flexbox grid:

- ➢ Bootstrap 4 makes easier to implement vertical designs, and the columns and rows can easily be implemented. the classes justify-

center-content can directly be used to align according to the requirement.

➢ Bootstrap 5 advanced grid system is made available , also columns don't have relative positions.

Ans: "Button Groups" in Bootstrap is a class of name "btn-group" which is used to create a series of buttons in groups (without spaces) vertically or horizontally. Syntax: This is the basic syntax of the button group class where each button has its own class of "btn"

A Button Group in Bootstrap is a component that groups together a set of buttons, allowing them to appear connected visually. This grouping can be horizontal or vertical, presenting a series of related actions or choices in a more cohesive manner.

Example:

<div class="btn-group" role="group" aria-label="Basic example">

  <button type="button" class="btn btn-primary">Button 1</button>

  <button type="button" class="btn btn-primary">Button 2</button>

  <button type="button" class="btn btn-primary">Button 3</button>

</div>

Ans:

In Bootstrap, you can create thumbnails—a smaller version of an image or content—with the help of the .card component or the .img-thumbnail class. Here's how you can use Bootstrap to create thumbnails:

To create thumbnails using Bootstrap −

Add an <a> tag with the class of .thumbnail around an image.

This adds four pixels of padding and a gray border.

On hover, an animated glow outlines the image.

Ans: Flexbox. The biggest difference between Bootstrap 3 and Bootstrap 4 is that Bootstrap 4 now uses flexbox, instead of floats, to handle the layout. The Flexible Box Layout Module, makes it easier to design flexible responsive layout structure without using float or positioning.

Flexbox offers a set of CSS properties that enable developers to create complex layouts with ease by allowing flexible box containers and their items to adjust dynamically based on available space and screen size. Some of the key properties associated with Flexbox include:

display: flex: This property is applied to the container element to enable the Flexbox layout. It converts the container's children into flexible items.

flex-direction: Defines the direction of the flex container's main axis. It can be set to row, row-reverse, column, or column-reverse.

justify-content: Determines how items are aligned along the main axis of the flex container. It controls the spacing and distribution of items. Values include flex-start, flex-end, center, space-between, and space-around, among others.

align-items: Specifies how items are aligned along the cross-axis of the flex container. It allows for alignment at the start, end, center, baseline, or stretch.

flex-grow, flex-shrink, flex-basis: These properties control how flex items grow, shrink, and set their initial size within the flex container.

Ans: Provide contextual feedback messages for typical user actions with the handful of available and flexible alert messages.

lerts are available for any length of text, as well as an optional close button. For proper styling, use one of the eight required contextual classes (e.g., .alert-success). For inline dismissal, use the alerts JavaScript plugin.

Example:

```
<div class="alert alert-success" role="alert">
  A simple success alert—check it out!
</div>
<div class="alert alert-danger" role="alert">
  A simple danger alert—check it out!
</div>
<div class="alert alert-warning" role="alert">
  A simple warning alert—check it out!
</div>
<div class="alert alert-info" role="alert">
  A simple info alert—check it out!
</div>
```

• **What is a bootstrap card and how would you create one?**

Ans: To create Bootstrap cards, you need to understand the card class and its customizations and layouts. A basic Bootstrap card consists of a . card class with the . card-body class, allowing you to add text, images, links, and other Bootstrap elements within a rounded border.

Example:

Images:

.card-img-top places an image to the top of the card. With .card-text, text can be added to the card. Text within .card-text can also be styled with the standard HTML tags.

```
<div class="card" style="width: 18rem;">
  <img src="..." class="card-img-top" alt="...">
  <div class="card-body">
    <p class="card-text">Some quick example text to build on the card title and make up the bulk of the card's content.</p>
  </div>
```