Web Designing Assignment

Module (Bootstrap Basic & Advanced) – 6

1. What are the advantages of Bootstrap?

Advantages	Description
Speedy Development	Bootstrap’s ready-made components accelerate development, enabling faster project completion.
Responsive by Design	The built-in grid system ensures websites adapt seamlessly to diverse screen sizes and devices.
Consistent UI Elements	Bootstrap maintains a uniform look and feel across various elements, ensuring a cohesive user experience.
Cross-Browser Compatibility	Bootstrap handles browser differences, minimizing the need for extensive testing and ensuring a consistent user experience.
Flexible Customization	Customizable themes, variables, and mixins provide flexibility to match specific project requirements and branding.
Community Support	Being open-source, Bootstrap benefits from a robust community, offering resources, documentation, and third-party plugins.

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

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. Basically, there are three types of container classes available in bootstrap:

1. Default-Container(container)
2. Responsive-Container(along with sm, md, lg, xl, xxl)
3. Fluid-Container(container-fluid)

1. Default-Container: The default container uses '.container' class. It provides a responsive fixed-width container.

Syntax:

```
<div class="container">

  <!-- Content here -->

</div>
```

2. Responsive-Container: Responsive containers are responsive in nature. Responsive containers allow you to specify a class that is wide until the specified breakpoint is reached, after which we apply max-widths for each of the higher breakpoints.

Syntax:

```
<div class="container-sm">Wide for small breakpoint</div>

<div class="container-md">Wide for medium breakpoint</div>

<div class="container-lg">Wide for large breakpoint</div>

<div class="container-xl">Wide for extra large breakpoint</div>

<div class="container-xxl">Wide for extra large breakpoint</div>
```

3. Fluid-Container: Fluid-container uses the '.container-fluid' class. It is used for a full-width container, spanning the entire width of the viewport.

Syntax:

```
<div class="fluid-container">

  <!-- Content here -->

</div>
```

Include Bootstrap and jQuery CDN into the <head> section before all other stylesheets to load our CSS.

```
<link rel=”stylesheet”
href=”https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css”
integrity=”sha384-
Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm”
crossorigin=”anonymous”>
<script src=”https://code.jquery.com/jquery-3.2.1.slim.min.js” integrity=”sha384-
KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN”
crossorigin=”anonymous”></script>
<script src=”https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js”
integrity=”sha384-
ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q”
crossorigin=”anonymous”></script
<script src=”https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js”
integrity=”sha384-
JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYl”
crossorigin=”anonymous”></script>
```

3. What are the default Bootstrap text settings?

Bootstrap is a popular front-end framework that provides a set of CSS and JavaScript components for building responsive web applications. The default text settings in Bootstrap are as follows:

- Font family: The default font family in Bootstrap is “Helvetica Neue”, Helvetica, Arial, sans-serif.
- Font size: The default font size for the body element in Bootstrap is 16px.
- Font weight: The default font weight for headings in Bootstrap is bold.

- **Line height**: The default line height for text in Bootstrap is 1.5.
- **Text color**: The default text color in Bootstrap is black (#000).
- **Text alignment**: The default text alignment in Bootstrap is left-aligned.
- **Text decoration**: The default text decoration in Bootstrap is none.
- **Text transform**: The default text transform in Bootstrap is none.

Note that these default text settings can be customized by modifying the relevant CSS classes or by using a custom CSS file.

4. What do you know about the Bootstrap Grid System?

Bootstrap's grid system allows up to 12 columns across the page.

If you do not want to use all 12 column individually, you can group the columns together to create wider columns:

<table>
<tr><td>span 1</td><td>span 1</td><td>span 1</td><td>span 1</td><td>span 1</td><td>span 1</td><td>span 1</td><td>span 1</td><td>span 1</td><td>span 1</td><td>span 1</td><td>span 1</td></tr>
<tr><td colspan="4">span 4</td><td colspan="4">span 4</td><td colspan="4">span 4</td></tr>
<tr><td colspan="4">span 4</td><td colspan="8">span 8</td></tr>
<tr><td colspan="6">span 6</td><td colspan="6">span 6</td></tr>
<tr><td colspan="12">span 12</td></tr>
</table>

Bootstrap's grid system is responsive, and the columns will re-arrange depending on the screen size: On a big screen it might look better with the content organized in three columns, but on a small screen it would be better if the content items were stacked on top of each other.

Tip: Remember that grid columns should add up to twelve for a row. More than that, columns will stack no matter the viewport.

❖ Grid Classes

The Bootstrap grid system has four classes:

- xs (for phones - screens less than 768px wide)
- sm (for tablets - screens equal to or greater than 768px wide)
- md (for small laptops - screens equal to or greater than 992px wide)
- lg (for laptops and desktops - screens equal to or greater than 1200px wide)

The classes above can be combined to create more dynamic and flexible layouts.

Tip: Each class scales up, so if you wish to set the same widths for xs and sm, you only need to specify xs.

❖ Grid System Rules

Some Bootstrap grid system rules:

- Rows must be placed within a .container (fixed-width) or .container-fluid (full-width) for proper alignment and padding
- Use rows to create horizontal groups of columns
- Content should be placed within columns, and only columns may be immediate children of rows
- Predefined classes like .row and .col-sm-4 are available for quickly making grid layouts
- Columns create gutters (gaps between column content) via padding. That padding is offset in rows for the first and last column via negative margin on .rows
- Grid columns are created by specifying the number of 12 available columns you wish to span. For example, three equal columns would use three .col-sm-4
- Column widths are in percentage, so they are always fluid and sized relative to their parent element

❖ Basic Structure of a Bootstrap Grid

The following is a basic structure of a Bootstrap grid:

```
<div class="container">
    <div class="row">
    <div class="col-*-*"></div>
    <div class="col-*-*"></div>
    </div>
    <div class="row">
    <div class="col-*-*"></div>
    <div class="col-*-*"></div>
    <div class="col-*-*"></div>
    </div>
    <div class="row">
    ...
    </div>
</div>
```

So, to create the layout you want, create a container (<div class="container">). Next, create a row (<div class="row">). Then, add the desired number of columns (tags with appropriate .col-*-* classes). Note that numbers in .col-*-* should always add up to 12 for each row.

5. What is the difference between Bootstrap 4 and Bootstrap 5

BASIS OF	BOOTSTRAP 4	BOOTSTRAP 5
Grid System	It has 5 tier (xs, sm, md, lg, xl).	It has 6 tier (xs, sm, md, lg, xl, xxl).
Color	It has limited colors.	Extra colors added with the looks, A card improved color palette. there are various shades available to choose.
Jquery	It has jquery and all related plugins.	Jquery is removed and switched to vanilla JS with some working plugins
Internet Explorer	Bootstrap 4 supports both IE 10 and 11.	Bootstrap 5 doesn’t support IE 10 and 11.

BASIS OF	BOOTSTRAP 4	BOOTSTRAP 5
Form elements	Radio buttons, checkboxes have different look in different OS and browsers. The form uses whatever default browsers provide.	The look of form elements will not change, on different OS or browser. The forms can be customized and form controls can be added, they would not depend on browser.
Utilities API	We cannot modify utilities in bootstrap 4	Bootstrap 5 gave freedom to modify and also create our own utility
Gutter	We use .glutter with fontsize in px	We use .g* with fontsize in rem
Vertical Classes	Columns can be positioned relative	Columns cannot be positioned relative
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
Jumbotron	It supports.	It doesn’t support jumbotron.
Card deck	The card deck is used to create a set of cards with equal width and height.	Card deck class in removed in bootstrap
Navbar	We have inline-block property and we will get white dropdown as default for dropdown-menu-dark class.	Inline-block property is removed and we will get black dropdown as default for dropdown-menu-dark class.
Static Site Generator	Bootstrap 4 uses Jekyll software.	Bootstrap 5 uses Hugo software as it is fast static site generator.

BASIS OF	BOOTSTRAP 4	BOOTSTRAP 5
flexbox grid	this 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.	advanced grid system is made available , also columns don’t have relative positions.
RTL Support	It does not enable RTL(Right to Left) switching.	It enables RTL(Right to Left) switching.
Offcanvas Component	It does not support Offcanvas Component.	It supports Offcanvas Component(that is it is available now).

6. What is a Button Group, and what is the class for a basic Button Group?

“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”.

```
<div class="btn-group">

  <button type="button" class="btn">Click</button>

</div>
```

Buttons have a default border-radius on the first and last buttons of the group.

Adding Styles on Buttons: Bootstrap allows you to add styles to your buttons using the following classes:

- .btn-default
- .btn-primary

- .btn-success
- .btn-info
- .btn-warning
- .btn-danger
- .btn-link

7. How can you use Bootstrap to make thumbnails?

Bootstrap helps web developers to create thumbnails that are used to show linked images in grids with the pre-defined classes which help to reduce codes length. Thumbnails are created to provide a quick preview of images with small images.

Thumbnail Image: A thumbnail is a small image that represents a larger image. Bootstrap has an easy way to do this with thumbnails. Bootstrap's *.thumbnail* class is used to show linked images in grids (grid system), a thumbnail is created using class *.thumbnail* within the element <a>. The class *.col-sm-** and *.col-md-** (where * represent number), it is used to create grids of the images.

Step by step guide for the implementation:

Step 1: Include Bootstrap and jQuery CDN into the <head> tag before all other stylesheets to load our CSS.

```
<link rel="stylesheet"
href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
<script
src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>
```

Step 2: Add <div> tag in the HTML body with class row. In that <div> create four *div* sections to create four images.

Step 3: Add "col-sm-6" and "col-md-3" to four *div* sections which creates webpage responsive.

Step 4: Add <a> tag with class value *thumbnail* to define the link of the image in the next line.

```
<a href="#" class="thumbnail">
```

8. In Bootstrap 4, what is flexbox?

In Bootstrap 4, flex box is used to control the layout and alignment specification of Bootstrap 4 components. This box makes it easier to design flexible responsive layout structures without using float or positioning attributes.

Let's take the following examples of Bootstrap 4 FlexBox.

```
<! DOCTYPE html>
   <html lang="en">
   <head>
    <title> Bootstrap 4 Flex Box Example </title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
   <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"> </script>
   <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js">
   </script>
   <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"> </script>
   </head>
   <style>
   h2 {
     position: relative;
     padding: 10;
     margin: 10;
     font-family: "Raleway", sans-serif;
     font-weight: 300;
     font-size: 40px;
     color: white;
     text-align: center;
     -webkit-transition: all 0.4s ease 0s;
     -o-transition: all 0.4s ease 0s;
     transition: all 0.4s ease 0s;
   }
```

```
body {
   background-size: 400% 400%;
   margin: 2rem;
   background: linear-gradient(-45deg, #ee7752, #e73c7e, #23a6d5, #23d5ab);
   animation: gradient 15s ease infinite;
   height: 100vh;
}
@keyframes gradient {
   0% {
      background-position: 0% 50%;
   }
   50% {
      background-position: 100% 50%;
   }
   100% {
      background-position: 0% 50%;
   }
}
</style>
<body>
<div class="container mt-3">
  <h2> Bootstrap 4 Flexbox Auto Margins </h2>
  <div class="d-flex mb-3 bg-secondary">
    <div class="p-2 mr-auto bg-info"> Flexbox box 1</div>
    <div class="p-2 bg-warning"> Flex box 2 </div>
    <div class="p-2 bg-primary"> Flex box 3 </div>
  </div>
  <div class="d-flex mb-3 bg-secondary">
    <div class="p-2  bg-info"> Flex box 1 </div>
    <div class="p-2 bg-warning"> Flex box 2 </div>
    <div class="p-2 ml-auto bg-primary"> Flex box 3 </div>
  </div>
</div>
</body>
</html>
```

9. How can one create an alert in Bootstrap?

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

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

This is a primary alert—check it out!

This is a secondary alert—check it out!

This is a success alert—check it out!

This is a danger alert—check it out!

This is a warning alert—check it out!

This is a info alert—check it out!

This is a light alert—check it out!

This is a dark alert—check it out!

```
<div class="alert alert-primary" role="alert">
  This is a primary alert—check it out!
</div>
<div class="alert alert-secondary" role="alert">
  This is a secondary alert—check it out!
</div>
<div class="alert alert-success" role="alert">
  This is a success alert—check it out!
</div>
<div class="alert alert-danger" role="alert">
  This is a danger alert—check it out!
</div>
<div class="alert alert-warning" role="alert">
  This is a warning alert—check it out!
</div>
```

```
<div class="alert alert-info" role="alert">
  This is a info alert—check it out!
</div>
<div class="alert alert-light" role="alert">
  This is a light alert—check it out!
</div>
<div class="alert alert-dark" role="alert">
  This is a dark alert—check it out!
</div>
```

10. What is a bootstrap card and how would you create one?

A **card** is a flexible and extensible content container. It includes options for headers and footers, a wide variety of content, contextual background colors, and powerful display options. If you're familiar with Bootstrap 3, cards replace our old panels, wells, and thumbnails. Similar functionality to those components is available as modifier classes for cards.

Example:

Cards are built with as little markup and styles as possible, but still manage to deliver a ton of control and customization. Built with flexbox, they offer easy alignment and mix well with other Bootstrap components. They have no margin by default, so use spacing utilities as needed.

Below is an example of a basic card with mixed content and a fixed width. Cards have no fixed width to start, so they'll naturally fill the full width of its parent element. This is easily customized with our various sizing options.

```
<div class="card" style="width: 20rem; margin : auto;">
  <img src="profile1.jpg" class="card-img-top" alt="...">
  <div class="card-body">
    <h5 class="card-title">Richard Taylor</h5>
    <p class="card-text">Software Engineer</p>
    <a href="#" class="btn btn-primary">View Profile</a>
  </div>
</div>
```