



Knowledge Check | Bootstrap Basics, Utilities, Forms, and Components

OVERVIEW: This assignment will assess your understanding of Bootstrap concepts covered in Module 5. You'll apply Bootstrap utilities, forms, tables, images, and components to create a fully functional, responsive web page.

INSTRUCTIONS: In VS Code, complete the following tasks in your `.html` and `.css` files.

Create a New Webpage Using Bootstrap:

1. Set up a basic HTML file with the Bootstrap **CDN** linked in the `<head>` section.
2. Use a **container** or **container-fluid** to structure your layout.

Part 1: Form Creation and Layout

Build a **Registration Form** using Bootstrap's form components:

- Include fields for **First Name**, **Last Name**, **Email**, **Password**, and a checkbox for agreeing to terms.
- Use **Bootstrap's grid layout** to position the **First Name** and **Last Name** fields side-by-side.
- Apply **validation classes** to ensure the **Email** and **Password** fields are required.
- Add a **Submit button** styled with `btn-success`.

Part 2: Table for Displaying Data

Simulate Data Collection:

- Below the form, create a **table** using Bootstrap's **table component**.
- **Hard-code user data** into the table to simulate submissions.
- Add **striped and hoverable rows** for better readability.
- Make the table **responsive** by wrapping it in a `table-responsive` div.

Part 3: Image and Button Utilities

Add an Image Section:

- Include a **responsive image** using the `img-fluid` class.
- Place the image inside a **fluid container** to stretch it across the page.
- Add a **circular image** (`rounded-circle`) below the main image.

Create Two Buttons with Utility Classes:

- One **button** should be **visible on all screen sizes**.
- The second **button** should be **hidden on small screens** using responsive display utilities (`d-none` and `d-md-block`).

Part 4: Navigation Bar

Build a Responsive Navigation Bar:

- Create a **navbar** with links to **Home**, **About**, and **Contact** sections.
- Ensure the **navbar collapses** into a **hamburger menu** on smaller screens.
- Use Bootstrap utilities and components to **style the navbar**.

Project Requirements

1. Form Creation and Layout:

- Use **Bootstrap's form components** to create a registration form with fields for **First Name**, **Last Name**, **Email**, **Password**, and a checkbox for agreeing to terms.
- Apply **Bootstrap grid layout** to position the **First Name** and **Last Name** fields side-by-side.
- Use **validation classes** to ensure the **Email** and **Password** fields are required.
- Add a **submit button** styled with `btn-success`.

2. Table for Displaying Data:

- Create a **Bootstrap table** to display user data.
- **Hard-code sample data** into the table to simulate user submissions.
- Apply **striped and hoverable rows** to enhance readability.

- Make the table **responsive** by wrapping it in a `table-responsive` div.
- 3. **Image and Button Utilities:**
 - Add a **responsive image** using the `img-fluid` class and place it inside a **fluid container** to stretch across the page.
 - Below the main image, add a **circular image** (`rounded-circle`).
 - Create two **buttons**:
 - One button should be **visible on all screen sizes**.
 - The second button should be **hidden on small screens** using responsive utilities (`d-none` and `d-md-block`).
- 4. **Responsive Navigation Bar:**
 - Build a **navbar** with links to **Home**, **About**, and **Contact** sections.
 - Ensure the **navbar collapses** into a **hamburger menu** on smaller screens.
 - Use **Bootstrap utilities** and components to style the navbar.
- 5. **Responsiveness and Layout:**
 - Use **Bootstrap containers** (`container` or `container-fluid`) to structure the layout.
 - Ensure that all components are **responsive** across different screen sizes using **media queries** and Bootstrap utilities.



Submission Guidelines

1. **GitHub Repository:**
 - Create a **GitHub repository** for this assignment.
 - Add your **HTML file** and any supporting CSS or images to the repository.
 - Include a **README.md** file with a brief description of your project, outlining key features and components used.
2. **Testing:**
 - Verify that the **navbar collapses** correctly on smaller screens.
 - Ensure all **buttons**, **images**, **forms**, and **tables** are styled as required.
 - Check that the page **renders properly across devices** and browsers.
3. **File Structure:**
 - Ensure all files are **organized** and named appropriately.
 - Place your **HTML file at the root** of the repository and organize any images or assets in appropriate folders.
4. **Submission:**
 - Submit the **GitHub repository link** in Google Classroom.
5. **Bonus Consideration:**

- Use **code comments** to highlight bonus features, making them easy to identify during grading.



Grading Criteria (Out of 25 Points):

Criteria	Points	Description
Form Creation	6 pts	Proper use of form components, grid layout, and validation.
Table Styling	5 pts	Responsive table with hard-coded data and proper styling.
Image Utilities	4 pts	Use of <code>img-fluid</code> , <code>rounded</code> , and <code>rounded-circle</code> .
Button Utilities	4 pts	Proper use of display utilities for responsive buttons.
Navigation Bar	5 pts	Navbar collapses correctly and is styled with Bootstrap.
Responsiveness & Layout	1 pt	Page adjusts correctly across different screen sizes.