

GIT Repo URL:

<https://github.com/prateekp1304/FSD/tree/main/FSD%20Assignment%202%2C3>

Roll No: 08

Name: Prateek Patel

PRN: 1032210532

FSD Laboratory 02

Aim: Design and develop a responsive web page using Bootstrap front end framework.

Objectives:

1. To understand HTML tags
2. To learn the styling of web pages using CSS
3. To learn Bootstrap Front end Framework.

Theory:

1. Bootstrap Grid System.

The Bootstrap Grid System is a responsive, mobile-first layout system that is part of the Bootstrap framework, a popular front-end framework for web development. It provides a structured way to create grid-based layouts for web pages that adapt to various screen sizes and devices, such as desktops, tablets, and smartphones. The Bootstrap Grid System is based on a 12-column grid, and it's designed to make web page layout and design more consistent and manageable.

2. Bootstrap .container and .container-fluid class.

In Bootstrap, the **.container** and **.container-fluid** classes are used to create container elements that help control the layout and spacing of the content within your web page. These classes are fundamental for structuring your content in a Bootstrap-based web application.

.container:

- The **.container** class creates a fixed-width container that centers its content horizontally within the browser window.
- It's ideal for creating layouts with a fixed maximum width, providing a consistent and controlled appearance for your content.
- The container's width adjusts responsively to different screen sizes, making it suitable for various devices, but it doesn't span the full width of the viewport.

.container-fluid:

- The **.container-fluid** class creates a full-width container that spans the entire width of the viewport.
- It's useful for creating full-width, edge-to-edge designs or when you want your content to take up the entire available horizontal space.

- Unlike `.container`, which has a fixed maximum width, `.container-fluid` adapts to the screen size and fills the entire viewport width.

FAQ:

1. What is a responsive website?

A responsive website is a website that is designed and developed to provide an optimal viewing and user experience across a wide range of devices and screen sizes, from large desktop monitors to small smartphones. The primary goal of responsive web design is to ensure that the website's content, layout, and functionality adapt and adjust automatically to accommodate the specific characteristics of each device or screen.

Key features and principles of responsive web design include:

- **Flexible Layouts:** Responsive websites use flexible grid layouts that adjust the placement and sizing of elements (such as text, images, and navigation menus) based on the screen size and orientation. This allows content to reflow and rearrange itself as needed.
- **Media Queries:** Media queries are CSS techniques that enable web designers to apply different styles and rules to a webpage based on the device's screen width, height, resolution, and other characteristics. Media queries are used to define breakpoints where the layout or styling changes to adapt to different screen sizes.
- **Fluid Images and Media:** Images and media elements (videos, maps, etc.) are typically set to scale proportionally within their containing elements, ensuring that they don't overflow or become too small to view on smaller screens.
- **Mobile-First Design:** Many responsive web designs adopt a "mobile-first" approach, where the initial design and development focus on creating a mobile-friendly layout. As the screen size increases, additional design enhancements are added for larger screens. This approach ensures that the website functions well on small screens and progressively enhances the experience on larger devices.
- **Touch-Friendly Design:** For mobile and touchscreen devices, responsive design often includes touch-friendly navigation elements, larger tap targets, and gestures to improve user interaction.
- **Content Prioritization:** On smaller screens, it's crucial to prioritize content to ensure that the most important information is readily accessible. Content may be reorganized or hidden behind menus or accordions to save space.
- **Cross-Browser Compatibility:** Responsive design aims to ensure compatibility with a variety of web browsers, including older versions, to reach the broadest possible audience.

- **Performance Optimization:** Optimizing website performance, such as minimizing the use of large images and scripts, is essential for responsive design, as it can impact load times, especially on mobile devices with slower connections.
- **Testing Across Devices:** Responsive websites are thoroughly tested on various devices and screen sizes to ensure that they function correctly and maintain a consistent user experience.

2. How Bootstrap helps to design a responsive website?

Here are several ways Bootstrap helps in designing a responsive website:

Responsive Grid System:

- Bootstrap's responsive grid system is based on a 12-column layout that can be customized to fit various screen sizes.
- It allows you to create complex, responsive layouts by defining how many columns an element should span at different breakpoints (e.g., mobile, tablet, desktop).
- The grid system ensures that your content rearranges and resizes appropriately on different devices.

Pre-designed CSS Classes:

- Bootstrap provides a wide range of pre-designed CSS classes that you can apply to HTML elements to style them responsively.
- Classes for typography, spacing, alignment, and more ensure consistent and visually appealing designs on all devices.

Responsive Navigation Components:

- Bootstrap offers responsive navigation components like responsive navigation bars (navbar), nav tabs, and navigation menus that automatically adapt to different screen sizes.
- Navbar components collapse into a mobile-friendly menu on small screens, improving user experience on mobile devices.

Responsive Images and Media:

- Bootstrap includes classes for responsive images, ensuring that images scale appropriately within their containers and that they don't overflow or become too small on different screens.

Mobile-First Approach:

- Bootstrap follows a mobile-first design philosophy, where the default styles and layouts are optimized for smaller screens. This approach ensures that your website looks good and functions well on mobile devices and progressively enhances the design for larger screens.

Built-in CSS Media Queries:

- Bootstrap uses CSS media queries to define breakpoints for different screen sizes, making it easy to customize the layout and styling at specific screen widths.

JavaScript Plugins:

- Bootstrap includes JavaScript plugins for various interactive components, such as modals, carousels, tooltips, and more, which are designed to work smoothly on all devices.

Utility Classes:

- Bootstrap provides utility classes for quickly adding responsive behavior to elements. For example, you can use classes like **.d-none**, **.d-md-block**, or **.mx-auto** to control display and spacing based on screen size.

Responsive Typography:

- Bootstrap includes classes for responsive typography, making it easy to adjust font sizes and line heights for different screen sizes.

Community and Documentation:

- Bootstrap has a large and active community, extensive documentation, and numerous third-party resources and themes, making it easy to find help and inspiration for your responsive design projects.

Output: Screenshots of the output to be attached.

Problem Statement:

Design and develop a responsive web page (For example student registration, course enrollment, library management system, online shopping system etc.) using Bootstrap front end framework. Web pages should contain HTML5 elements (Use all possible formatting for example font, colour etc.). Use all possible formatting for example,

- font,
- colour etc.

Web page should include various

- images,
- links within the page,
- tables
- lists
- form elements
- paragraphs
- links to other pages for navigation,
- new tabs.

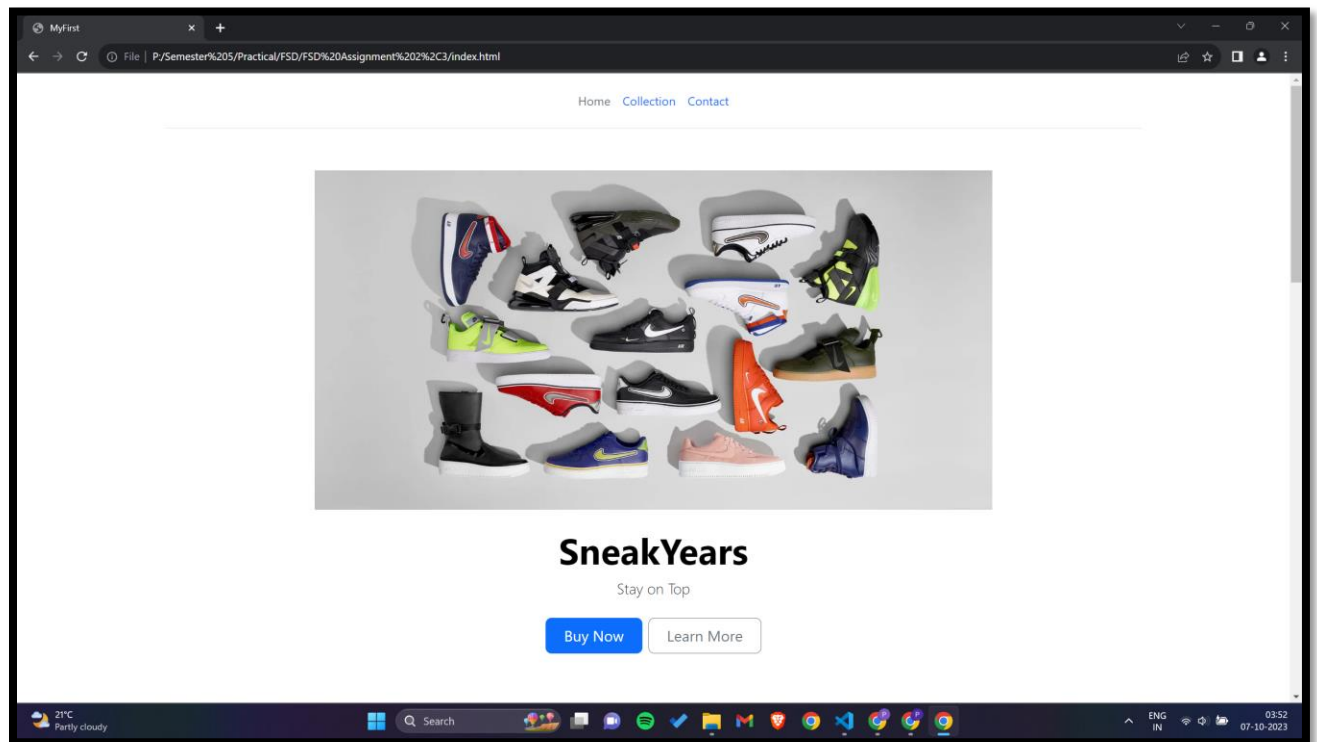
Use Cascaded Style Sheets (CSS) to style the web pages designed. Make use of

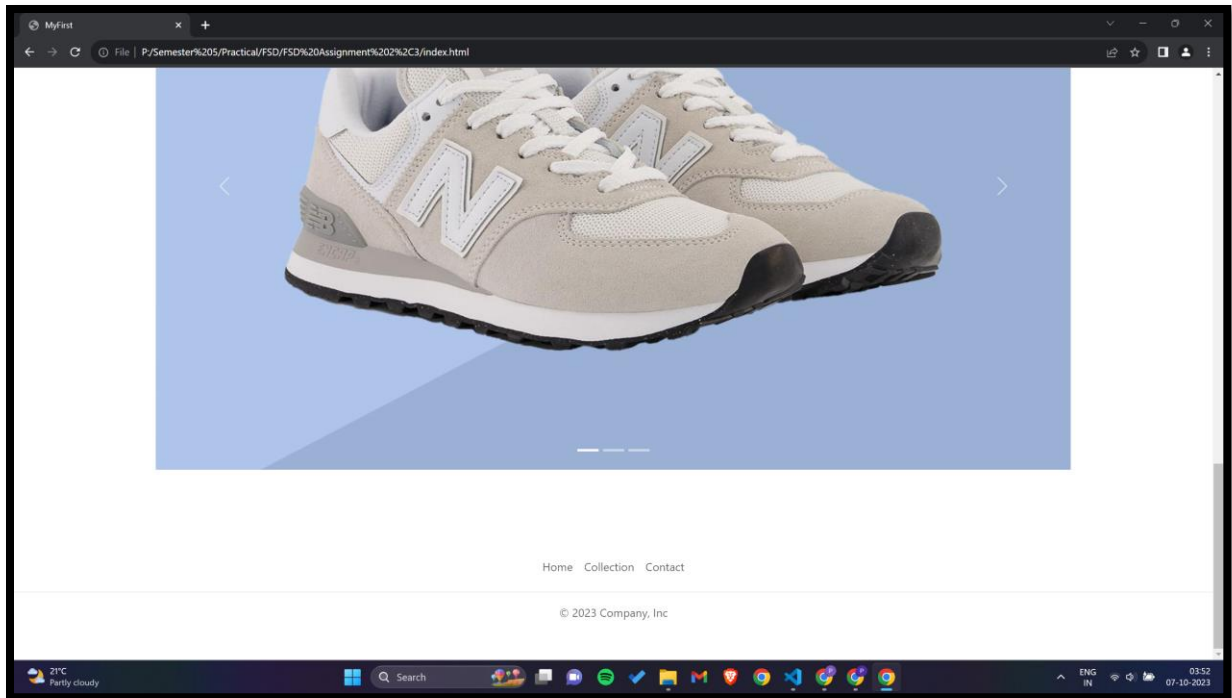
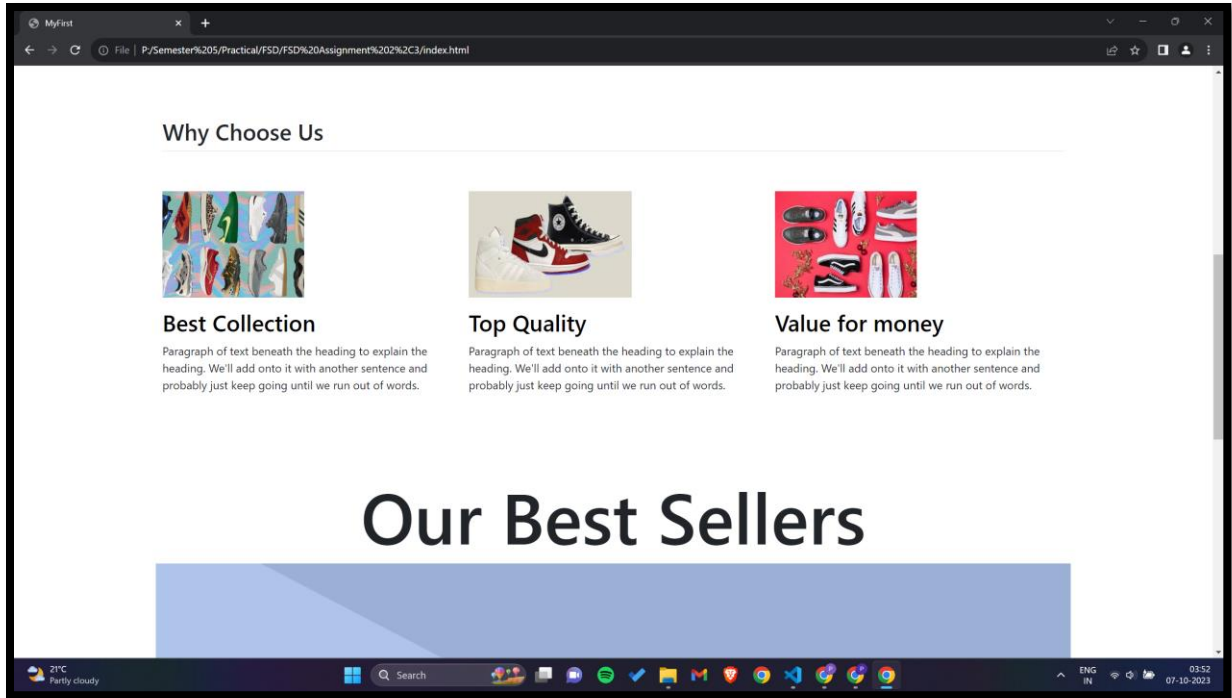
- Three types of styles (Inline, Internal and External)
- border,
- margins,
- padding,
- navigation,

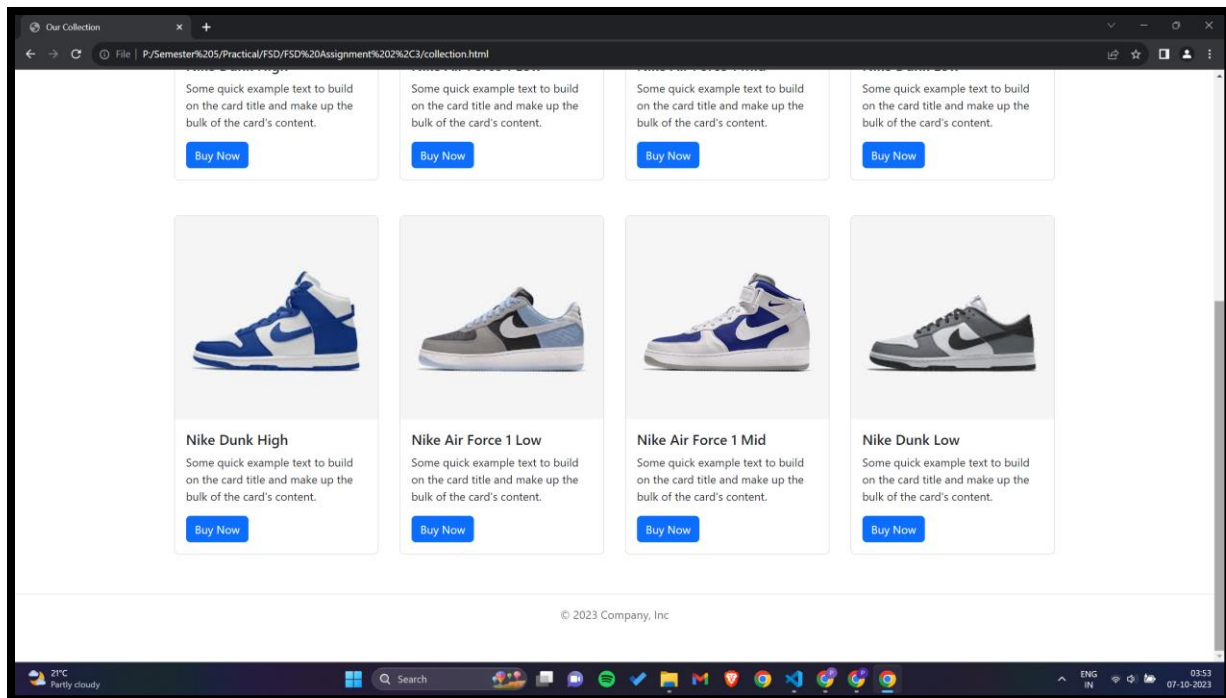
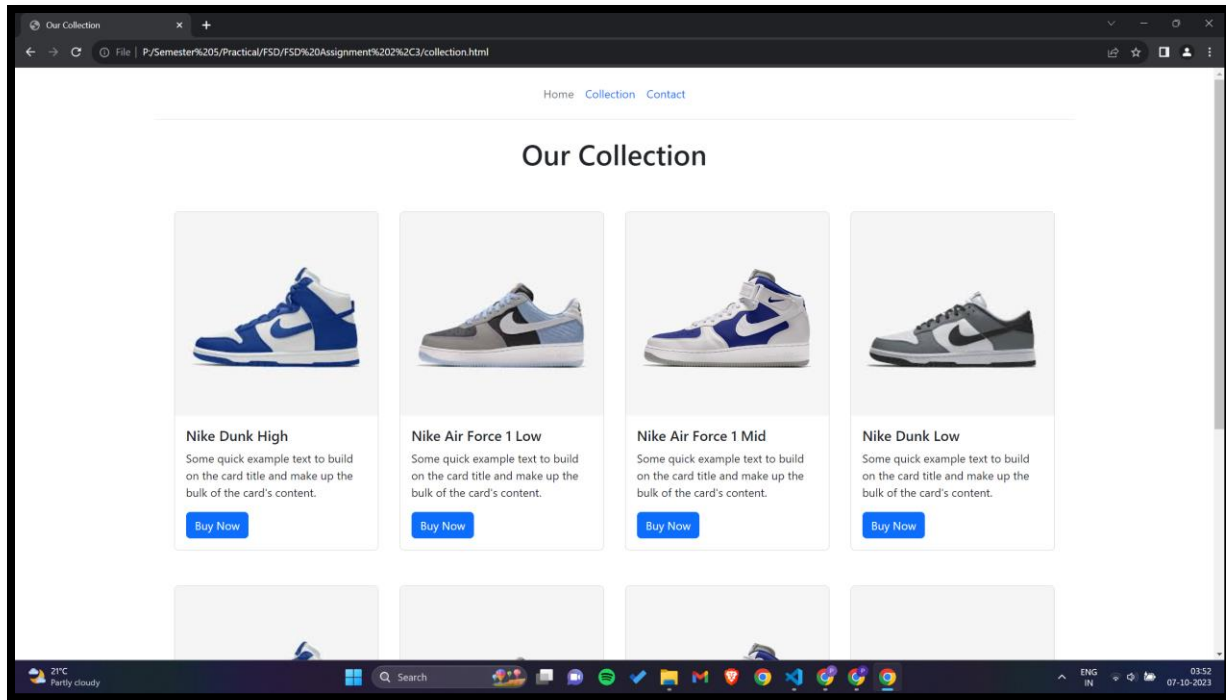
- dropdown list
- tag selector
- class selector
- id selector
- exploring background image and position property etc.

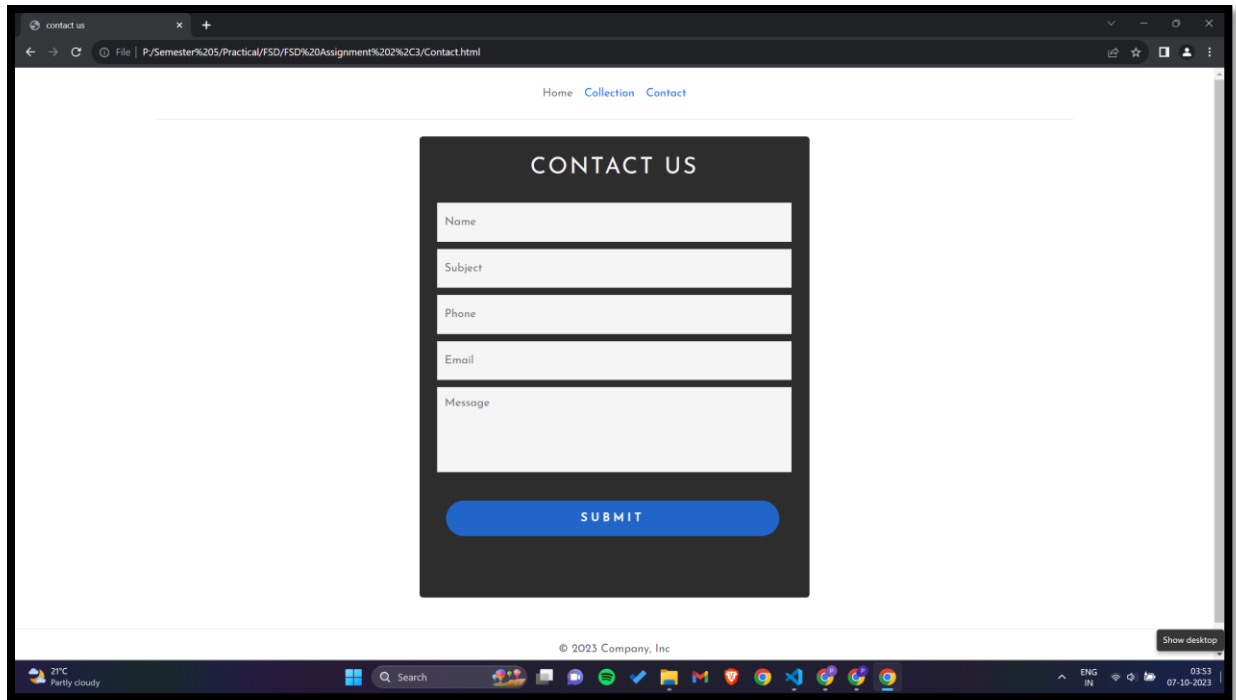
Output Screenshots:

:









The screenshot shows a web browser window with a dark theme. The address bar displays the URL: `File | P:\Semester%205\Practical\FSD\FSD%20Assignment%20%2C3>Contact.html`. The page has a navigation bar with links: [Home](#), [Collection](#), and [Contact](#). The main content is a dark gray contact form titled "CONTACT US". The form contains the following fields: "Name", "Subject", "Phone", "Email", and "Message". Below these fields is a blue "SUBMIT" button. The footer of the page reads "© 2023 Company, Inc". The Windows taskbar at the bottom shows the date and time as 07-10-2023, 03:53, and the language as ENG IN.

contact us

Home Collection Contact

CONTACT US

Name

Subject

Phone

Email

Message

SUBMIT

© 2023 Company, Inc

21°C
Partly cloudy

Search

ENG
IN

03:53
07-10-2023

Show desktop