

# FSD Assignment - 1

Niharika Sonalkar

1032232922

Roll no. 62

Aim: develop responsive web designs using HTML5, containing a form, style the pages using CSS.  
Use of Tag selector class selector and id selector

Objectives:

1. To understand HTML tags.
2. To learn styling of web pages using CSS.
3. To learn Bootstrap front end framework

Theory:

1) Responsive Web Design: (RWD)

- It is an approach to the web design that makes web pages well on a variety of devices and window or screen sizes. This layout adapts automatically to the screen size orientation and platform, ensuring usability and aesthetics.
- Primary goal: provide an optimal meaning and intended user experience: easy reading and navigation with minimal resizing, panning & scrolling.

2) Role of `<meta name="viewport">`

- tag tells the browser how to control the page dimensions
- Role: sets the visible area of a webpage to match the decision screen and initial zoom level.
- Essential for RWD: without this tag, mobile browser assume a default viewpoint width, which causes the page to appear zoomed out & not scale to small

### 3) Bootstrap & Grid System.

- Bootstrap helps create responsive website that adopt devices (mobile, tabl., desktop).
- Uses a 12-column grid system where content is placed inside row & col elements.
- the grid adapts using breakpoints with prefixes:
  - col-sm → extra small ( $< 576\text{px}$ )
  - col-sm → small ( $\geq 576\text{px}$ )
  - col-md → medium ( $\geq 768\text{px}$ )
  - col-lg → large ( $\geq 992\text{px}$ )

### 4) Difference between Tag, Class & ID selectors.

Selectors	Symbol	Purpose	Example
→ Tag	none	styles all <code>&lt;p&gt;</code>	<code>p { color: red; }</code>
→ Class	.	styles all elements highlighted	<code>.bg-yellow { background-color: yellow; }</code>
→ ID	#	styles specific	<code>#main { font-size: 20px; }</code>

### 5) Three Ways to Apply CSS

i) Inline CSS: written directly in HTML elements

`<p style="color: blue;"> Hello </p>`

ii) Internal CSS: written inside a `<style>` tag in `<head>`

`<style>`  
`p { color: green; }`  
`</style>`

iii) External CSS: stored in a CSS file & linked in `<link rel='stylesheet' href='style.css'>`

## Problem Statement:

Build a responsive web page the (single-page website for a simple survey.

### Conclusion:

In this assignment, we learned how to build responsive web pages using HTML 5, CSS and Bootstrap. I understood the use of different CSS selectors, the ways to apply CSS, and how the viewpoint tag with Bootstrap's grid system ensure layouts adapt to all screen sizes.

By

# FSD Assignment - 2

Niharika Sonalkar

1032232922

Roll no. 62.

Aim: Develop a web application using Javascript to implement sessions, cookies, DOM. Perform validation such as checking for emptiness, only numbers for phone.

Objectives:

1. To understand what form validation is.
2. To learn basic functioning of DOM objects.
3. To learn how to apply various techniques to implement it.

Theory:

1) Role of Regular Expression

Regular expressions (regex) are patterns used to match and manipulate strings. They are ideal for validating data formats like phone numbers, emails, or passwords because they can define strict rules for structure (e.g.: 10 digits for a phone number). This ensures input follows a consistent format before being processed.

2) Session vs. Cookie

A cookie is stored in the user's browser and holds small amounts of data (like session IDs).

A session is stored on the server and tracks

user activity through the session ID shared via cookies. Together, they maintain a logged-in state - cookies store the identifier while sessions store the actual user data securely on the server.

### 3) Client-side vs Server-side Validation.

Client-side validation gives quick feedback to users but can be bypassed if someone disables JavaScript or manipulates requests. Server-side validation ensures rules are enforced securely before processing. If you rely only on client-side validation, a hacker could send malicious data (like SQL injection) directly to the server.

```
4) <p id = "msg"> Before submission </p>
<button onclick = "change msg 11">
Submit </button>
<script> function change Msg () {
document.getElementById("msg").innerHTML = "form submitted"; </script>
```

### 5) Steps for Frontend to MySQL connectivity.

- Frontend (HTML / CSS / JS): Collect user input through forms.
- Backend (PHP / Node.js): Process form data & connect to MySQL using libraries.

- Database (MySQL) : Run SQL queries.  
(INSERT, SELECT, UPDATE).
- Response : Backend sends results back to frontends (like success / fetched data)
- Display : Frontend updates the page with the response dynamically.

FAQs :

Q1 Write 3 reasons why Form validations are important:

- Ensures only correct and complete data is submitted.
- Prevents SQL injection, XSS, and other security attacks.
- Reduces server load by catching errors early on the client side.
- Improves user experience with instant feed back.
- Maintains data integrity in the database.

Q2 Give an Example of how to modify an attribute value using DOM.

→ <a id="my link" href="old.html">  
click here </a>

```
<script>
document.getElementById("my link").  
setAttribute("href", "newpage.html");
```

Q3 What are different features of Java Script?

- • lightweight, interpreted, and case-sensitive scripting language.
- Supports functional, object-oriented, and event-driven programming.
- Provides dynamic typing (no need to declare variable types).
- Allows DOM manipulation for interacting and dynamic webpages.
- cross-platform, supported by all major browsers.

Q3  
diagrams

# FSD Assignment - 3

Niharika Sonalkar

1032232922

Roll no. 62

Aim:

Design a responsive, interactive photo carousel using react that demonstrates components, state & props.

Objectives:

- Build a component-based UI using React.
- Manage dynamic state and accept props.
- Handle user events and ensure accessibility.

Theory:

1. Roles of State and Props:

State holds internal, mutable data for a component while props are read-only values passed from parent to child. Props enable reusability and state manages internal behaviour.

2. What is a React Component? Class vs. Functional.

→ A component is reusable piece of UI. Class components use life cycle methods and

this state, while functional components use hooks for managing state and life cycle. Functional components are simpler, cleaner and preferred due to hooks.

### 3. Templating using components in React.

This means breaking UI into reusable pieces like buttons, cards, or carousels instead of one monolithic HTML file. It improves maintainability, scalability and reusability compared to traditional web development.

### 4. Handling user events in React

User interactions like clicks are handled by attaching handlers. This updates state using hooks causing the components to re-render with updated data.

### 5. Responsive web design

Responsive design ensures the UI adapts to various screen sizes. In react, that can be achieved using CSS media queries, responsive frameworks or CSS-in-JS libraries, ensuring a consistent experience across.

## Problem Statement:

Create a responsive interactive photo carousel application using React.

## Conclusion:

The React Photo carousel successfully demonstrates component-based architecture, state and props management.

BY  
31/10/26

# FSD Assignment - 4

Niharika Sonalkar

1032282922

Roll no. 62.

Aim: Enhance web page developed in earlier assignment by rendering lists and portals. Error handling, Routes and style with React CSS also make it a responsive design.

Objective :

- Enhance UI and UX.
- Improve application Robustness and Navigation.

Theory :

1. How do lists and keys work in React?

→ Lists in React are used to render multiple elements dynamically. Each element in a list should have a unique key prop to help React identify the changes.

2. What is a React Portal and When do we use one?

→ A react portal allows rendering a component's children to a different part of the DOM outside the main component tree.

It is useful for modals, tooltips or dialogs that need to visually break out of their parent container.

### 3. Discuss the importance of Error Boundaries in React.

Error boundaries catch JavaScript errors anywhere in the component tree. It is useful for improving the Robustness and UI. It displays a fallback UI instead of crash.

### 4. How does React Router enable Single Page App (SPA) functionality?

React Router manages navigation without actually reloading the page by mapping URL paths to components. This enables SPA behaviour where different views are rendered dynamically within the same app.

### 5. Explain the different ways to style a React app.

Styling in React can be done using CSS stylesheets, inline styles, CSS modules, CSS-in-JS libraries or utility-first framework in Tailwind CSS.

Problem Statement:

Create a dynamic user dashboard with multiple sections that can be navigated using tabs.

Conclusion:

In this lab, the webpage was enhanced using React features like lists, Portals, Error boundaries and React Router along with styling. The app became more interactive, robust and responsive.

21/10/26

# FSD Assignment - 5

Niharika Sonalkar

1032232922

Roll no. 62

## Aim:

Develop a responsive web design using Express framework to perform CRUD operations and deploy with Node JS. Use MongoDB.

## Objectives:

- Develop a full-stack Web application.
- Demonstrate Backend Development & Deployment

## Theory:

1. What is the role of Express.js as a web framework for Node.js?

Express.js simplifies server-side development by providing features for routing, middleware and handling HTTP requests. It helps build scalable web apps efficiently.

2. Explain the concept of CRUD operations.

CRUD stands for Create, Read, Update and Delete. These are the fundamental operations that allow apps to manage & manipulate persistent data in database.

3. Why is MongoDB a suitable choice for this project?

MongoDB is a NoSQL database that stores data in flexible JSON-like documents, making it easy to handle dynamic and unstructured data. It integrates well with Node.js and supports high scalability.

4. What steps are involved in deploying a Node.js and Express apps?

Deployment typically involves setting up a server environment, installing dependencies, connecting the app to a database, configuring environment variables, and hosting the app using services like Heroku, AWS or Vercel.

Problem Statement:

Develop a recipe-sharing platform with CRUD operations using Express.js and MongoDB.

Conclusion:

This lab demonstrated full-stack development by implementing CRUD operations with Express.js and MongoDB, resulting in a responsive and functional web app.