

Assignment - 01

Name : Darshil Patel
Roll No : 41
PRN : 81032232467

Aim : Develop responsive web design using HTMLs, containing a form, style the page using CSS, Use of tag Selector, class Selector and id Selectors. Use Inline, Internal and External CSS, Apply Bootstrap CSS.

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] Responsive Web Design (RWD)

- It is an approach to web design that makes web pages render 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 across desktop, tablets and smartphones.

- Primary Goal : provides an optimal meaning and interaction experience, easy reading and navigation with minimal resizing, Panning and scrolling across wide range of devices

2] Role of <meta name="viewport">

- <meta name="viewport"> tags tell the browser how to control the page dimensions and scaling.
- Role : It sets the visible area of a web page to match the device screen width and initial zoom level.
- Essential for RWD without this tags, mobile browser assume a default newport width, which cause the page to appear zoomed out and not scale to small screen.

3] Bootstrap & Grid system

- Bootstrap helps create responsive website that adapts devices (mobile, table, desktop)
- Uses a 12-column grid system where content is placed inside .row and .col element
 - the grid adapts using breakpoints with prefixes.
 - .col -> extra Small ($< 576 \text{ px}$)
 - .col-sm -> Small ($> 576 \text{ px}$)
 - .col-md -> medium ($> 768 \text{ px}$)
 - .col-lg -> large ($> 992 \text{ px}$)
 - .col-xl -> extra large ($\geq 1200 \text{ px}$)

4] Difference between Tag, class & ID Selectors

Selectors Symbol Purpose Examples

- Tag None Styles all `<P>` tags `P {color: red}`
- class Styles all elements with given class `.highlight {background-color: yellow}`
- ID `#` Styles one specific element with id `#main {font-size: 20px}`

5] Three ways to Apply CSS

- 1) Inline CSS: written directly in HTML elements
`<P style="color: blue;>Hello</P>`
- 2) Internal CSS: written inside `<style>` tag in `<head>`

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

3) External CSS: Stored in a CSS file and linked in
<link rel='stylesheet' href='style.css'>

Problem Statement .

4. Blog Post layout

Conclusion :

- In this assignment, I learned how to build responsive web pages using HTML 5, CSS and Bootstrap.
- understood the use of different CSS Selectors, the ways to apply CSS and how the viewport tag with Bootstrap grid system ensures layouts adapt to all screen sizes.

CH/25
12/8/25

Assignment - D 2

1032232467

Aim : Develop a web application using javascript to implement sessions, cookies, DOM. Perform Validation such as checking for emptiness, only numbers for phone numbers, special character requirement for password, regular expressions for certain format of the fields etc. Use the mysql database.

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] Explain the role of regular expressions. why th are they a suitable tool for validating data formats like a phone number or checking for the presence of specific characters in a password ?
- Regular expressions are patterns used to match and validate text formats. They are suitable for validating data like phone numbers or passwords because they allow precise rules - Such ensuring a phone number contains only digits or a password include special characters. This helps prevent invalid data entry and improves input accuracy.

- 2] Explain the fundamental difference between a Session and a cookie in the context of web application development. How do they work together to maintain a user's logged-in state?
- A cookie is a small piece of data stored on the client's browser, which can persist across visits and is sent with every request to the server.
- A session stores user data securely on the server and is identified by a unique session ID.
- When a user logs in, the server creates a session and sends the session ID to the client as a cookie. The client's browser stores this cookie and includes it in subsequent requests. The server uses the session ID from the cookie to retrieve the user's session data, thereby maintaining the user's logged-in state. This separation ensures sensitive data is kept on the server while the client holds only the identifier.
- 3] What is the purpose of performing both client-side and server-side validation? Describe a scenario where relying solely on client-side validation could lead to a security vulnerability.
- Client-side validation provides quick feedback to users and reduces server load, while server-side validation ensures data security and integrity because it cannot be bypassed. Relying only on client-side validation is risky since users can disable or manipulate it. For example, without server-side checks, an attacker could submit malicious input like SQL injection, leading

to security breaches.

- 4] Provide a simple example of how a JavaScript script can interact with the DOM to dynamically change the content of a web pages after a user action, such as a form submission.
- when a user submits a form, JavaScript can capture the submit event, prevent the default page reload, and then update the webpage dynamically. For example, the script can retrieve the form's input value, hide the form and display a thank-you message.

```
<form id="myForm">
  <input type="text" id="name" required />
  <button type="Submit" > Submit </button>
</form>
<div id="message"></div>
<script>
  const form = document.getElementById('myForm');
  const message = document.getElementById('message');

  form.addEventListener('Submit', function(event) {
    event.preventDefault();
    const name = document.getElementById('name').value;
    message.textContent = `Hello, ${name}! Thank you
    for Submitting`;
    form.style.display = 'none';
  });
</script>
```

5] Give the steps for connectivity from front end using HTML CSS JS to mysql.
→ Steps to connect frontend

- ① Frontend (HTML + CSS + JS)
 - Create a form in HTML
 - Use JS (fetch or AJAX) to send form data to backend.
- ② Backend (PHP / Python etc)
 - Receive the request
 - Connect to MySQL using driver
 - Run SQL queries (INSERT, SELECT, etc)
 - Send back response (Success/Error).
- ③ MySQL Database
 - Create database and tables
 - Store / retrieve the data requested.

FAQs

1. Write 3 reasons why form validation are important
→ 1. Data Accuracy & Quality - ensures users enter correct and complete data (e.g. valid email, required fields).
 2. Security - Prevents malicious inputs like SQL Injection XSS or Script injections.
 3. Better User Experience - Gives instant feedback reducing errors and frustration.
2. Give an example of how to modify an attribute value using DOM.

```
<!DOCTYPE html>
<html>
<head>
<title> DOM Example </title>
</head>
<body>

<button onclick="changeImage()">
```

```
<script>
function changeImage() {
let img = document.getElementById("myImage");
img.setAttribute("src", "new.jpg");
}
</script>
</body>
</html>
```

when the button is clicked, the src attribute of the image changes from old.jpg to new.jpg.

3. What are the different features of JavaScript?
 - 1. Lightweight & interpreted - Run directly in the browser without compilation.
 - 2. Event-Driven :- Responds to events like clicks, mouse moves & key press .
 - 3. Cross Platform - works in almost all browsers and operating systems .

- A. Rich Built-in functions For math, string, date array, and dom manipulation
- S. Object - Oriented - Supports objects, inheritance and reusable code classes.

~~89/90~~

Object oriented programming is a way of organizing code by defining objects and classes and objects interact with each other.

Object

Object is a combination of data and methods. It is a collection of properties and methods grouped together. It is a self contained unit.

Object is a class instance.

Object is a variable that holds an object.

Object is a variable that holds an object. It is a self contained unit. It has properties and methods. It is a collection of properties and methods grouped together. It is a self contained unit.

Object is a variable that holds an object. It is a self contained unit. It has properties and methods. It is a collection of properties and methods grouped together. It is a self contained unit.

FULL STACK DEVELOPMENT

Assignment - 03

Page No.

Date

Aim: Design an interactive front end application using React by implementing templating using Components, states and Props, classes, Events. It must be responsive to scale across different platforms.

Objectives:

To develop a responsive, interactive front-end application using React.js that effectively demonstrates the fundamental concepts of component-based architecture, state management and event handling. The application will serve as a practical exercise in building a scalable user interface by implementing templating with components, managing dynamic data with states and props, and handling user interactions with events, ensuring a seamless user experience across various devices and screen sizes.

Theory:

1. Explain the role of State and Props in React. How do they differ, and what is the primary purpose of each in managing data flow within a component-based application?

→ React application are built using components need a way to manage and pass data.

- State is the internal data of a component.
- It is mutable and controlled within the component.
- Updating state causes the component to render with new values.
- Example: Storing a counter value or form input.

- Props

- Props are external input passed from a parent component to a child component.
- They are immutable (read-only) inside the child component.
- Used to customize component and allow data flow from parent to child.
- Example: Passing a title to a Card Component

State

- Managed within Component - internal
- Mutable
- To store local, dynamic data
- To share data and customize components

Props

- Passed from Parent Component
- Immutable

Primary Purposes:

- State → Manage internal, dynamic data
- Props → Passes data from parent → child to enable reusability

2.

- What is a React component? Differentiate between a class component and a functional component and discuss the advantages of using a functional component with hooks like useState and useEffect
- A React component is a reusable UI block that accepts input(props) and returns UI.

Class Component

- Written as ES6 Class.
- Uses this.state and lifecycle methods

functional component

- Written as a function

- Use hooks like useState and useEffect

Advantage of functional + Hooks : Cleaner syntax, no this keyword, simpler state, and easier testing.

3. Describe the concept of templating using components

→ in React why is this approach considered superior to traditional web development methods that rely on monolithic HTML files?

→ React uses components as templates to build UI instead of one large HTML file.

• Each component can be reused with different props.

Superiority over traditional HTML:

1. Reusable and modular
2. Easier to maintain | debug
3. Dynamically updates UI when data changes
4. Promotes separation of concerns

4.

How do you handle user events in React? Provide a simple code snippet to demonstrate how an event handler is defined in a component and how it can be used to update the component's state
 → React uses synthetic events for handling user action like clicks, inputs etc.
 Event handlers are defined in the component and passed to elements.

```
import { useState } from "react"
function App() {
  const [msg, setMsg] = useState("Hello!");
  function handleclick() { setMsg("Button Clicked!"); }
  return (
    <div>
      <p>{msg}</p>
      <button onClick={handleclick}>Click Me!</button>
    </div>
  );
}
```

5. What is responsive web design? Why is it crucial for modern application? Describe how you would implement a responsive design in a React application using CSS media queries or a CSS-in-JS library.
- Responsive Web Design makes applications adapt to different screen sizes.
- Importance:
- provides consistent UX, improves accessibility and SEO.
 - Implementation in React:
 1. CSS media queries → defines styles for different screen widths
 2. CSS-in-JS libraries → write responsive styles directly inside React component

(B)
Q10/25

FULL STACK DEVELOPMENT

Page No.	
Date	

Assignment - 04

Aim : Enhance web page developed in earlier assignment by rendering Lists and Portals, Error Handling, Router and style with React CSS also make it a responsive design to scale well across pc, tablet and Mobile Phone.

Objectives :

- Enhance User Interface and Experience
- Improve Application Robustness and Navigation

Theory :

- 1] How do lists and keys work in React ?
→ Lists are used to render multiple elements dynamically by using "map()".
Each element should have a unique key to help React identify changes and optimize re-rendering.

Example :
 $\Sigma [1, 2, 3]. map((num) \Rightarrow <li key={num}>\Sigma num)$

- 2] What is a React Portal and when would you use one?
→ React Portal is a way to render children outside their Parent component's DOM heirarchy.

It is useful when you need to render components that are visually on functionally separate, like models, tool tips or popups, but want to keep them logically within the same Component structure.

3]

Discuss the importance of error boundaries in React.

→ Error boundaries are components that catch Javascript errors anywhere in their child component tree, log these errors and then finally display a fallback UI.

They prevent the entire application from crashing when an error occurs, providing a more user-friendly experience by sharing a graceful fall back instead of a broken UI.

4]

How does React router enable Single Page Application (SPA) functionality?

→ React Router enables SPA functionality by allowing you to define routes and navigate b/w different components without causing full-page reloads. It dynamically changes the view in response to URL changes, providing a seamless user experience within client-side routing.

5]

Explain the different ways to style a React application.

→ **Inline Styling:** Using the style attribute with object.

CSS Style Sheets: Importing regular CSS files & applying class names.

CSS Modules: Scoped styles using module.css files to avoid naming conflicts.

~~Conclusion:~~ This assignment has my React knowledge by adding lists, portals, error handling, routing and responsive styling, enhancing usability, navigation and reusability across devices.

Q1
Ans

Assignment - 05

Aim: Develop a responsive web design using Express framework to perform CRUD operations & deploy with Node.js. Use MongoDB.

- Objective:
- 1] Develop a full stack Web Application.
 - 2] Demonstrate Backend Development & Deployment Proficiency.

Theory:

- 1] What is the role of Express.js as a Web framework for Node.js?
 - Provides a lightweight fast & flexible web framework built on top of Node.js.
 - Simplifies handling HTTP requests & responses.
 - Supports routing.
 - Middleware support for request processing.
 - Enables building RESTful APIs & full stack application efficiently.
 - Reduces boilerplate code compared to using pure Node.js.
- 2] What is a React Portal & when would you use one?
 - A portal allows you to render a component's children into a DOM node that exists outside the parent component's hierarchy.
 - Syntax: `ReactDom.createPortal(child, container)`
 - Useful for:
 - Avoiding CSS overflow / Positioning issues.
 - Models, dialogs, tool tips, dropdowns.

3] Discuss the important of Error Boundaries in React

- Error Boundaries wrap components that catch JavaScript errors in their child component tree.

- They prevent the entire app from crashing when an error occurs.

- They can show a fallback instead of breaking the whole app.

- Crucial for production apps where stability & user experience matter.

4]

How does React Router enables Single Page Application (SPA)

functionalities?

- Read - Router manages navigation without reloading the page.

- It uses the History API to update the URL while keeping the app alive.

- Components are rendered conditionally based on the route, enabling seamless transitions.

5]

Explain the different ways to style a React application.

→ Inline styling → Using the style attribute with object.

CSS stylesheets → Importing regular CSS files & applying class names.

CSS modules → Scoped styles using module .css files to avoid naming conflicts.

(b) ~~Ques 8~~

Conclusion: In summary express.js simplifies backend development for node.js, while mongo DB offers flexible, scalable database solution. This helps to demonstrate both backend & deployment skills.