

AASHIAT KOLLURU  
PRN: 10B2231219  
Roll no.: 23  
TY, CSF

## FSD Lab Assignment 1 :

Aim : To develop responsive web design using HTML5, containing a form. Style pages using CSS, use tag selector, class selector & id selectors. Use inline, internal & external CSS. Apply bootstrap CSS.

Objectives : (i) Understand HTML tags

(ii) Learn styling of web pages using CSS.

(iii) Learn bootstrap front end framework

### Theory

Problem statement : Create single page portfolio showcasing your skills & projects. Page must include sections about me & contact. Design must be responsive, employing bootstrap's grid system. Styling should be implemented using mix of inline, internal & external CSS.

### Theory :-

① Define responsive web design (RWD). What is its primary goal?

→ Responsive web design is a web design approach that ensures website looks & functions well on all devices. Its primary goal is to create a flexible, user friendly interface that automatically adapts layout, images & other elements based on device's screen size.

② Explain role of the `<meta name="viewport"` tag. why is it essential for PWA?

→ The tag tells browser how to control page dimension & scaling.

`width = device-width` → width matches device screen width.

`initial-scale=1.0` → sets initial zoom = 100%. without it mobile may shrink or zoom out of pages.

③ How does bootstrap assist in creating responsive layout? discuss concept of grid system & how it adapts to different screen size.

→ Bootstrap helps create responsive layouts by providing 12 columns grid system. You place content inside columns & bootstrap automatically arranges them.

④ Differentiate b/w tag, class, ID selectors.

	Syntax	Applies to	Uniqueness
Tag selector	<code>p{color: blue;}</code>	All <code>&lt;p&gt;</code> elements	Not unique
Class selector	<code>.highlight{color: red;}</code>	All elements with <code>class = "highlight"</code>	Can be reused
ID selector	<code>#main{background-color: yellow;}</code>	Single element with <code>id = "main"</code>	Must be unique

- ⑤ describe three main ways to apply CSS to HTML.
- i) Inline CSS  
→ Applied directly to an element using style attribute
  - ii) Internal CSS.  
Written inside a <style> tag in HTML <head>
  - iii) External CSS.  
Stored in separate .css file & linked using  
<link rel = "stylesheet" href = "style.css">

Conclusion: with this assignment I got to learn to build responsive webpage with help of HTML, CSS tags.

21/8/25

Aim: develop a web application using JS to implement sessions, cookies, DOM. Perform validations such as checking for emptiness only no. for phone no. special character requirement for password, etc. Use MySQL database.

Objectives: (i) To understand what form validation is.  
(ii) To learn basic functioning of DOM objects.  
(iii) To learn how to apply various techniques to implement it.

### Theory:

#### ① Role of Regular expressions (RegEx)

→ RegEx are pattern for matching strings useful for validating formats like phone no., passwords or emails by enforcing rules on characters.

#### ② Explain fundamental difference b/w a session and cookie in context of web application development. How do they work together to maintain user's logged in State?

→ Cookies are small pieces of data stored on client side used to remember user preferences.  
→ Sessions store user data on server identified by a session ID usually stored in a cookie. Together, they maintain user login states securely by keeping sensitive data server side while using cookies for identification.

③ What purpose of performing both client side & server side validation? Describe scenario where relying solely on client side validation could lead to security vulnerability?

- Client side validation provide instant feedback & reduces server load.
- Server side validation can be bypassed by attackers leading to security risks such as SQL injection or data corruption.

④ Provide example how JS can interact with DOM to dynamically change context of a web page after user action, such as form submission.

```
<p id = "msg" > Hello </p >
<button onclick = "changeText()"> click here </button>
<script>
    function changeText() {
        document.getElementById("msg").innerHTML
        = "Text changed";
    }
</script>
```

⑤ Give steps for connectivity from frontend using HTML & JS to MySQL.

→ Steps for frontend to MySQL connectivity:

- Create HTML form to collect data.
- Use JS / AJAX to send data to server.
- Server side script (node.js, PHP, etc.) receives data.
- Server side script connects MySQL database.
- Insert or query data in MySQL.
- Send response back to front end.

#### FAQs:

① Write 3 reasons why form validations are important.

→ Three reasons why form validations are important:

(i) prevent submission of incomplete or incorrect data.

(ii) Improve user experience with instant feedback.

(iii) Enhance security by avoiding malicious input.

② Give example of how to modify attribute value using DOM.

→ `document.getElementById('myImage').setAttribute('src', 'newimage.jpg')`

③ What are different features of JS?

→ Client side scripting language

- dynamic typing
- Event driven programming
- DOM manipulation capability
- Supports OOPS & functional programming.

Conclusion:

This assignment demonstrates importance of client side form validation, efficient DOM manipulation using JS & integration of jQuery for enhanced interactivity.

✓  
By  
d19125

## Topic FSD assignment 3:

Date \_\_\_\_\_

Aim: Design an interactive front end application using React by templating using components state & props, class, events. It must be responsive to scale across diff platforms.

Objective: Develop a responsive, interactive front end app using React.js that effectively demonstrates component based architecture, state management with scalable UI components, dynamic data via states & props & seamless user interactions across devices.

Theory :

① State: Represent mutable data owned & managed by a component. It allows components to create dynamic & interactive UI by updating themselves when states changes.

props: short for properties, props are read only inputs passed from a parent component to a child enabling data sharing across components.

Difference: State is internal & changeable while props are external & immutable. Together, they enable unidirectional data flows in react.

② What is react component? Differentiate b/w class & functional component. Discuss adv. of using functional component with hooks.

→ Component: A reusable independent piece of UI in react.

Class component: defined using ES6 classes use this state & lifecycle methods.

Functional components: Defined as functions use hooks like useState & useEffect for state & life cycle management.

Advantages of functional & hooks: clearer syntax, less boiler plate, better performance, easier to test & modern react development favors hooks over classes.

③ Describe concept of "templating using components" in react. Why is this approach considered superior.

→ Breaks UI into reusable, modular components. Superior to monolithic HTML → improves reusability, scalability & maintainability.

④ How do you handle user events in react? provide a simple code snippet to demonstrate how event handler is defined in a component & how it can be used to

update components state.

→ Handlers defined as functions; update state using: use state

import {useEffect, useState} from "react";

function Counter () {

const [count, setCount] = useState(0);

return

<div>

<p> {count} </p>

<button onClick={() => setCount(count + 1)}> Inc

</button>

</div>

? ;

}

⑤ what is responsive design why is it crucial for modern applications? describe how you could implement responsive design in react application using CSS media queries or CSS in - JS library.

→ Ensures UI adapts to all screen sizes / devices Implement using CSS media queries or CSS in JS.

eg: @ media (max width: 600px) {  
div { font-size: 14px; }  
}

conclusion: development of user profile dashboard efficiently demonstrated use of core concepts such as state, prop & event handling; By using functional component & hooks this enhances code maintainability & provides seamless user experience.

(B) ~~2019/10~~

Aim: Enhance Develop a responsive web page developed in earlier assignment by rendering lists & portals, error handling, routers & style with react css also make responsive design to scale well across PC, mobile phone.

Objectives:

- Enhance User Interface & experience
- Improve application robustness & navigation.

### Theory :

① How do lists & keys work in react ?

→ Lists in react allow you to render multiple elements dynamically using JS's map() method to loop over data.

→ Keys are unique identifiers assigned to each element in a list to help react efficiently update & render only changed items during re-renders.

② What is a React portal & when would you use one ?

→ React portal is a way to render children outside their parent component's DOM hierarchy.

It is useful when you need to render components that are visually or functionally separate like modals, toolbars or pop-ups but want to keep them logically within same component structures.

③ Discuss importance of error boundaries in react.

- Error boundaries are components that catch javascript errors anywhere in their child component tree, log those errors & then finally display fallback UI.
- They prevent entire app from crashing when error occurs providing more user friendly experience by showing graceful fallback instead of 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 fullpage reloads. It dynamically changes views in response to URL changes providing a seamless UI with client side routing.

Q5) Explain different ways to style a react application.

- CSS style sheets: Traditional external or internal CSS files
- CSS in JS: Using libraries like styled components or emotion to write scoped CSS in JS
- Inline styles: Using style attribute with JS objects.
- CSS modules: Scoping CSS locally to components to avoid global style clashes.
- preprocessors: Using tools like SASS or LESS with react for enhanced CSS features.

Problem statement: Expand e-commerce product gallery application to include responsive modal for displaying product details.

Conclusion: This assignment has my react knowledge by adding hits, portals, error handling, routing & responsive styling, enhancing usability, navigation & reliability across devices.

BD  
23/9/18

Topic

# FSI Assignments :

Date

Aim: Develop a responsive web design using express framework to perform CRUD operations & deploy with Node JS using MongoDB.

Objective:

- Develop a fullstack web app.
- Demonstrate backend development

## Theory :

① What is role of express.js as web framework in Node.js ?

- Provides lightweight fast & flexible web framework on top of Node.js.
- Simplifies handling HTTP req & responses.
- Supports routing.
- Middleware support for req processing.
- Enables building RESTful API & full stack apps.
- Reduces boilerplate code compared to using pure Node.js.

② What is react portal & when would you use one ?

- A portal allows you to render a component's children into a DOM node that exists outside parent component's hierarchy.

Syntax : React DOM, Create portal (child, container)

useful for : . Avoiding CSS overflow / positioning issues.

. Models, dialogs, tooltips, dropdowns.

③ Discuss the importance of error boundaries in React.

- Error boundaries are react components that catch JS errors.  
10
- They prevent entire app from crashing when error occurs.
- They can show a fallback instead of breaking whole.
- Crucial for production apps where stability & user experience matter.

④ How does react router enable single page app.

→ React router manages navigation without reloading page.

✓ Use `useHistory` API to update URL while keeping app alive.

Components are rendered conditionally based on route enabling seamless transitions.

- Q) Explain diff. ways to style a react app:
- Inline styling + Using style attribute with obj
  - CSS stylesheet → Importing regular.css file & applying class names.
  - CSS modules → scoped styles using module.css files to avoid naming conflicts

Conclusion: In summary, express.js simplifies backend development for node.js, while mongoDB offers flexible, scalable database sol<sup>n</sup>. This helps to demonstrate both backend & deployment skills.

BN  
27/10/25