**Department of Electrical and Electronics Communication Engineering**

**Subject: Full Stack Development**

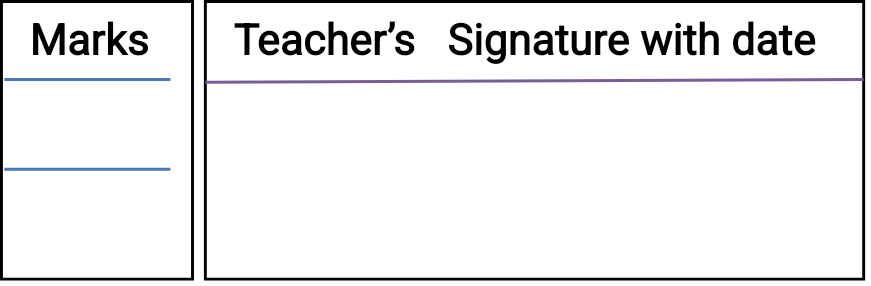
**Semester: VI AY 2025-26**

**Name: Rishabh Bajaj Class: T. Y. B. Tech.**

**Roll No: 1032221881 Batch: G2**

**Experiment No: 3**

**Name of the Experiment: JavaScript for PHP Sessions, Cookies, and DOM real-time update.**



**Performed on: 16/01/2025**

**Submitted on: 05/02/2025**



**Aim:** JavaScript for PHP Sessions, Cookies, and DOM real-time update.

**Objective:**

* Understanding how JavaScript interacts with PHP sessions and cookies for user authentication and data persistence.
* Implementing JavaScript techniques to read, create, and modify cookies for session management.
* Using AJAX to fetch and update PHP session data without refreshing the page.
* Dynamically updating the DOM in real time based on session or cookie changes.

**Platform:** VS code, PHP, Code runner extension

**THEORY:**

Instructions: you are requested to write answer of following questions in A4 size pages.

1. What are PHP sessions? And how JavaScript can access PHP session data?
2. What are cookies, and how do they differ from sessions? And how can JavaScript create, read, and delete cookies?
3. What is the Document Object Model (DOM), and how does JavaScript manipulate it?
4. How can JavaScript update the DOM based on changes in PHP session data?

**Problem Statement 1:**

* Write a program to design Student registration form by using HTML, CSS having following fields: Username, Email, Phone number, Password, Confirm Password and write external javascript code to achieve following validations
  + Fields should not be empty. If spaces are entered those should be considered empty
  + Phone number must accept only numeric values and it should be 10 digits
  + Password length must be at least 7 and it should contain at least one capital letter, one digit and one special character from the set (&,$,#@)
  + Value entered in password field and confirm password fields must match
  + Email address must contain @ sign and a ., there should be few letters before the @ sign, there should be three letters between @ sign and a . There must  be 3 or 2 letters after the  .   (hint: Use regular expression)

**Problem Statement 2:**

* Write a client-side script with JavaScript to access and manipulate Document Object Model (DOM) objects in an HTML web page. Develop a dynamic web page using javascript and DOM. Make use of the following for accessing elements
  + getElementById, getElementsByTagName,getElementsByClassName
  + Change the text using innerHTML property
  + Change the CSS properties like color, position of a particular element on the page
  + Change the image source after clicking on a button
  + Add a text node and attach it to a parent node
  + Delete a node
* Include jQuery to perform following operations:
  + Change button text using jQuery.
  + Set background-image using jQuery CSS property.
  + Access HTML form data using jQuery.
  + Add attribute using jQuery
* Use this reference link for jQuery: **https://www.w3resource.com/jquery-exercises/part1/index.php**

**Problem Statement 3:**

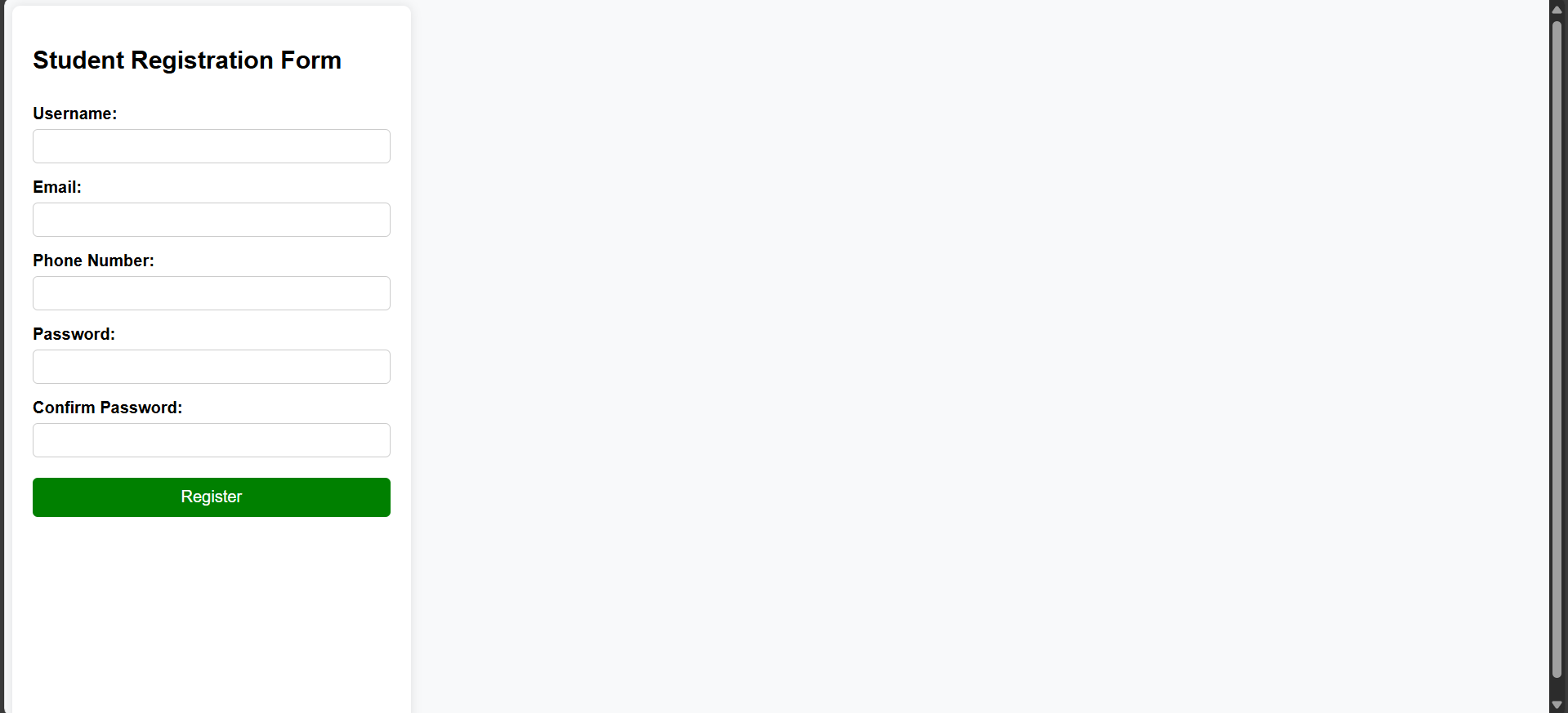
Real-Time User Activity TrackerScenario:

* Track user activity (e.g., last visited page, current time) using cookies and display it dynamically.
* Requirements:
  + Use PHP to store the user's last visited page and visit timestamp in a cookie.
  + Retrieve the cookie using JavaScript and display the information on the page.
  + Update the cookie whenever the user visits a new page.

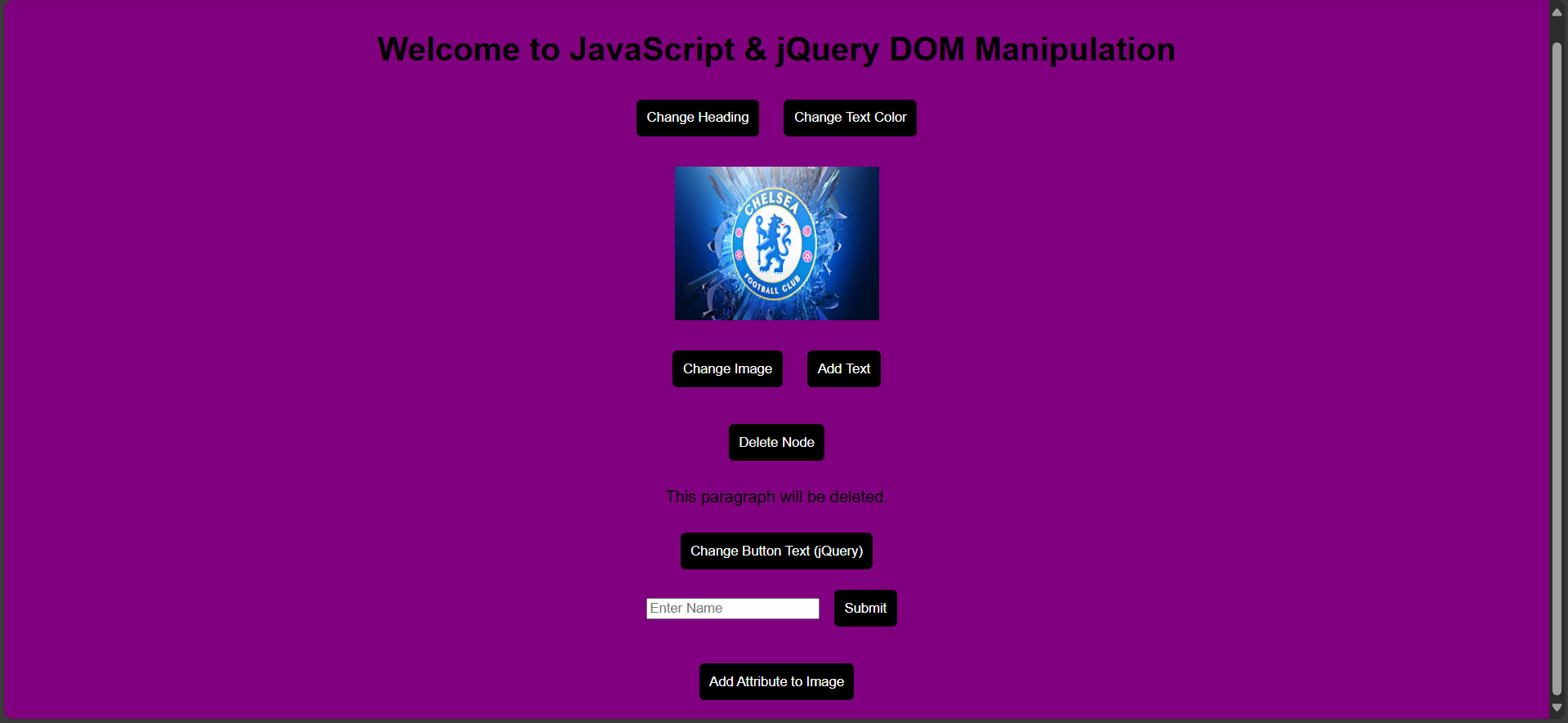
**Output:**

1. Screenshots of the output of problem statement to be attached.

Q1]



Q2]



Q3]



2. Share GitHub repository link of above project

**<https://github.com/RishabhBajaj002/FSD_Lab>**

**Conclusion:**

In this experiment, we explored the integration of JavaScript with PHP sessions and cookies to enhance user experience and maintain state across web pages. We also implemented real-time DOM updates using JavaScript, allowing dynamic content changes without requiring a full page reload. This approach improves interactivity, enhances performance, and ensures seamless data handling. Overall, the experiment demonstrated how JavaScript and PHP can work together to create more responsive and efficient web applications.

**Additional Links:**

<https://youtu.be/SbAfPJj0H4g?si=lzDgy3vcX_CwyV41>

<https://youtu.be/SbAfPJj0H4g?si=zVFp4SBAVMwAxz95>

<https://youtu.be/7zcXPCt8Ck0?si=S_oBHp_ErmlQPbCY>

**Post lab question:**

1. How can AJAX be used to retrieve PHP session data without reloading the page?
2. What is the difference between document.cookie in JavaScript and setcookie() in PHP?
3. What are some best practices for efficiently updating the DOM without causing performance issues?

