Lab Report: Web Programming Skills

Title: Interactive Web Features using JavaScript and CSS

Objective:

To implement dynamic and user-friendly web pages that enhance interactivity and provide a smooth user experience using HTML, CSS, and JavaScript.

Theory:

HTML (HyperText Markup Language) forms the backbone of web pages, defining their structure and content. CSS (Cascading Style Sheets) enhances the visual aesthetics, while JavaScript enables interactivity and dynamic behavior. The combination of these technologies is fundamental for building engaging and responsive web applications.

This report showcases two implementations:

1. A button to display the current date and time.
2. A login form with client-side validation.

Lab 2.1: Show Time Button

Objective:

To create a webpage with a button that dynamically displays the current date and time upon being clicked.

Implementation Details:

- **HTML Structure:**
  - Contains a heading and a button.
  - A `<div>` element to display the current date and time dynamically.
- **CSS Styling:**
  - Gradient background for aesthetic appeal.
  - Button styling includes hover effects for interactivity.
- **JavaScript Functionality:**
  - Event listener attached to the button to fetch and display the current date and time in a readable format.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Show Time Button</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            text-align: center;
            margin-top: 50px;
            background: linear-gradient(to right, #8d46da, #3c7ce9);
            color: #fff;
            display: flex;
            justify-content: center;
            align-items: center;
            flex-direction: column;
            height: 100vh;
            margin: 0;
        }
        button {
            padding: 15px 30px;
            background-color: #fff;
            color: #fc3325;
            border-radius: 5px;
            cursor: pointer;
        }
    </style>
</head>
<body>
    <h1>Click the button to Show the Current Time:</h1>
    <button id="showTimeBtn">Show Time</button>
    <div id="time"></div>
    <script>
        document.getElementById('showTimeBtn').addEventListener('click',
function() {
            const currentDate = new Date();
            const options = { weekday: 'long', year: 'numeric', month:
'long', day: 'numeric', hour: '2-digit', minute: '2-digit', second: '2-
digit', hour12: true };
            document.getElementById('time').innerText = `Current Date & Time:
${currentDate.toLocaleString('en-US', options)}`;
        });
    </script>
</body>
</html>
```

Discussion and Conclusion:

The implementation successfully demonstrates real-time data fetching and rendering using JavaScript. The visually appealing design enhances the user experience. Potential improvements could include additional formatting options or support for different locales.

Lab 2: Static Website Layout

Code

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Welcome to My Website</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 0;
            background-color: #f4f4f4;
            color: #333;
        }
        header {
            background-color: #4CAF50;
            color: white;
            padding: 15px 0;
            text-align: center;
        }
        nav {
            background-color: #333;
            text-align: center;
        }
        nav a {
            color: white;
            padding: 14px 20px;
            text-decoration: none;
        }
        nav a:hover {
            background-color: #ddd;
            color: black;
        }
        .main-content {
            padding: 20px;
            text-align: center;
        }
        footer {
            background-color: #333;
            color: white;
            text-align: center;
            padding: 10px 0;
        }
    </style>
</head>
<body>
    <header>
        <h1>Welcome to My Website</h1>
    </header>
    <nav>
        <a href="#">Home</a>
        <a href="#">About</a>
        <a href="#">Services</a>
        <a href="#">Contact</a>
```

```
    </nav>
    <div class="main-content">
        <h2>About This Website</h2>
        <p>This is a simple example of a basic HTML website layout.</p>
    </div>
    <footer>
        <p>&copy; 2024 My Website. All Rights Reserved.</p>
    </footer>
</body>
</html>
```

Discussion

The static website layout emphasized clean structure and design principles. By using semantic HTML tags and CSS for styling, the page achieved a professional look. The navigation menu and footer demonstrated reusable design patterns common in web development.

Lab 2.3: Login Form with Validation

Objective:

To create a simple login form with client-side validation to ensure that both username and password fields are filled before submission.

Implementation Details:

- **HTML Structure:**
  - A form containing input fields for username and password.
  - A button for submission.
- **CSS Styling:**
  - Clean and modern design with a centered layout.
  - Error messages styled for better visibility.
- **JavaScript Functionality:**
  - Validation logic to check if fields are empty and display appropriate error messages.

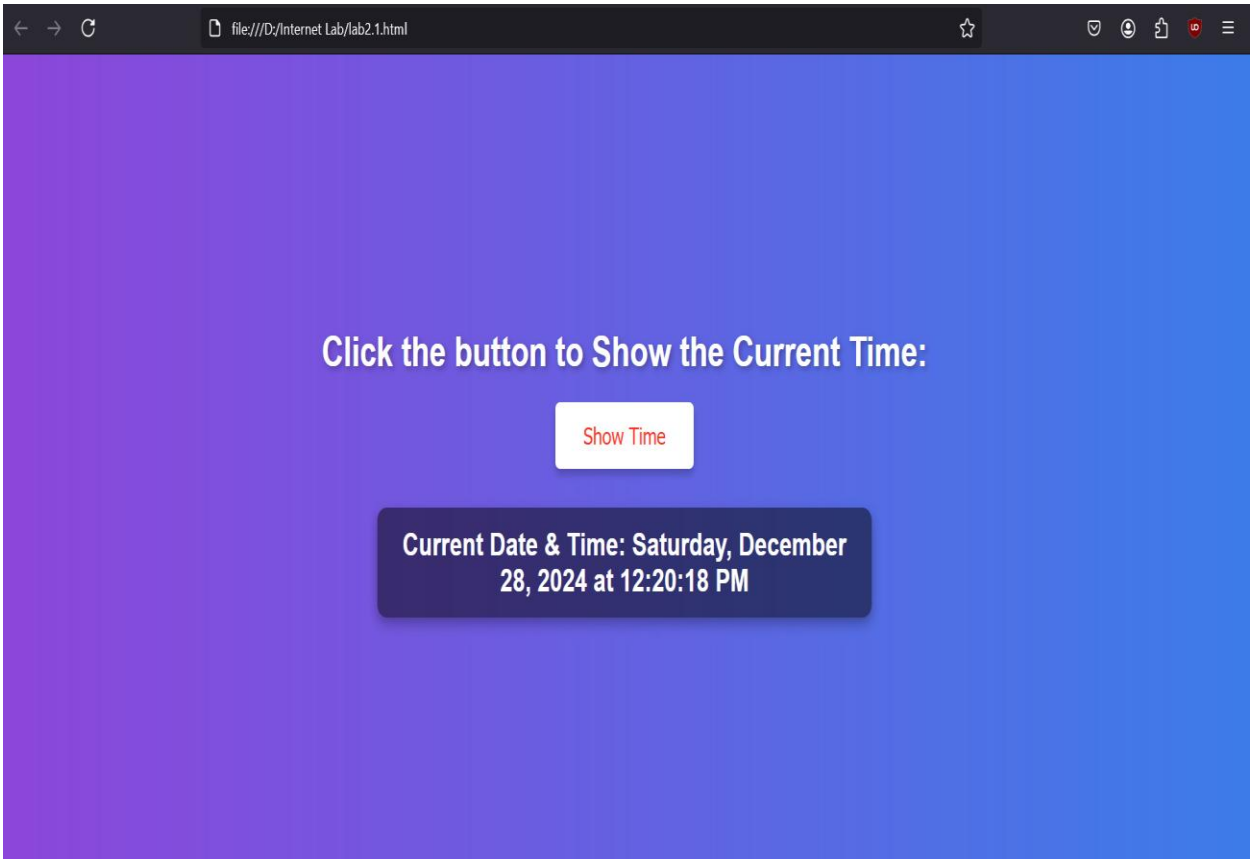
Code:

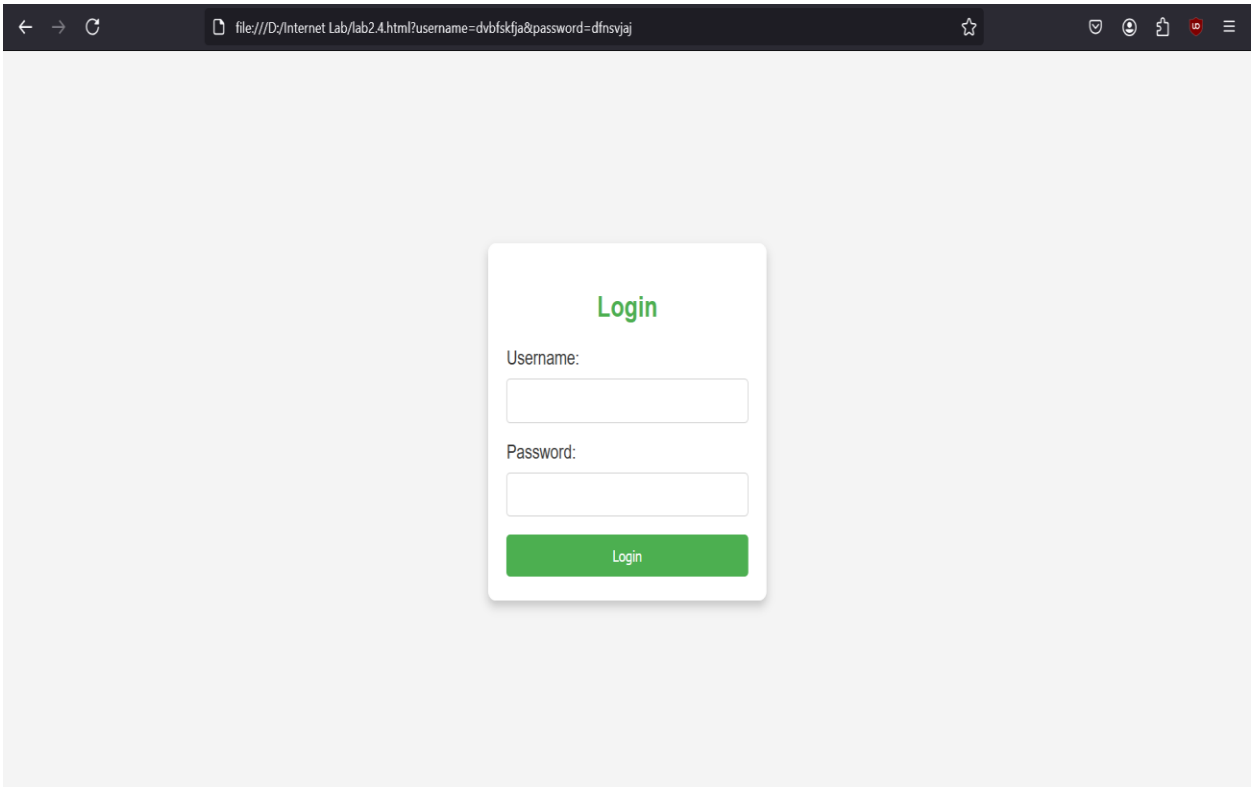
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Form Validation</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
```

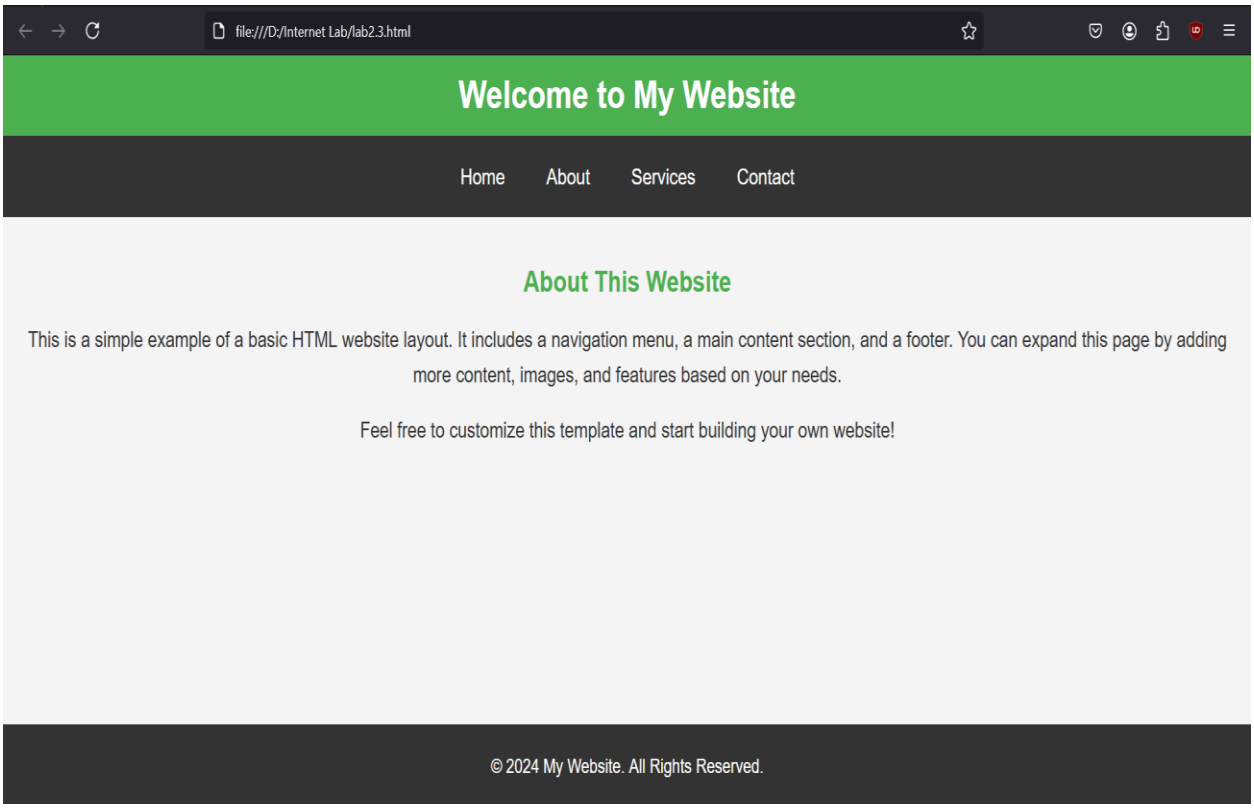
```
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            margin: 0;
        }
        .form-container {
            background-color: white;
            padding: 20px;
            border-radius: 8px;
            box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
        }
        .error {
            color: red;
            font-size: 14px;
            margin-bottom: 10px;
        }
    </style>
</head>
<body>
    <div class="form-container">
        <h2>Login</h2>
        <form id="loginForm" onsubmit="return validateForm()">
            <div class="error" id="error-message"></div>
            <label for="username">Username:</label>
            <input type="text" id="username" name="username">
            <label for="password">Password:</label>
            <input type="password" id="password" name="password">
            <button type="submit">Login</button>
        </form>
    </div>
    <script>
        function validateForm() {
            const username = document.getElementById('username').value;
            const password = document.getElementById('password').value;
            const errorMessage = document.getElementById('error-message');
            errorMessage.textContent = '';
            if (!username || !password) {
                errorMessage.textContent = 'Please enter both username and
password';
                return false;
            }
            return true;
        }
    </script>
</body>
</html>
```

Discussion and Conclusion:

This lab illustrates client-side validation, enhancing user experience by providing immediate feedback. The form design is simple yet effective. Further improvements could include password strength indicators and additional validation criteria.


file:///D:/Internet Lab/lab2.1.html
Click the button to Show the Current Time:
Show Time
Current Date & Time: Saturday, December 28, 2024 at 12:20:18 PM


file:///D:/Internet Lab/lab2.4.html?username=dvbfskfja&password=dfnsvjaj
Login
Username:
Password:
Login


file:///D:/Internet Lab/lab2.3.html
Welcome to My Website
Home
About
Services
Contact
About This Website
This is a simple example of a basic HTML website layout. It includes a navigation menu, a main content section, and a footer. You can expand this page by adding more content, images, and features based on your needs.
Feel free to customize this template and start building your own website!
© 2024 My Website. All Rights Reserved.