Web Technologies Laboratory 03

Aim: Client-side Form Validations using JavaScript, DOM real-time update, JQuery to develop Ajax based applications.

Aim: Write a program to perform following form validations using JavaScript:

a) All fields mandatory,
b) Phone number, Email Address, Zip code Validation etc.
Include JavaScript to access and manipulate Document Object Model (DOM) objects in an HTML web page.
Include JQuery to develop to develop your application as an Ajax based application.

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. Different types of form validations.
2. HTML Document Object Model.
3. What is JQuery? Write various JQuery Selectors.

Different Types of Form Validations

4. **Client-Side Validation:**
   - **Description:** Performed in the browser before the form is submitted to the server. It provides instant feedback to users, preventing submission of invalid data.
   - **Methods:**
     - **HTML5 Validation:** Built-in form attributes like `required`, `pattern`, `type`, `min`, and `max`.
     - **JavaScript Validation:** Custom scripts to validate input fields.
5. **Server-Side Validation:**
   - **Description:** Performed on the server after the form is submitted. It's a critical layer of security as it ensures that even if client-side validation is bypassed, the data is still validated before processing.
   - **Methods:**
     - **Server-Side Scripts:** PHP, Node.js, Python, etc., are used to validate data against defined rules.

HTML Document Object Model (DOM)

- **Description:** The HTML DOM is a programming interface that represents the structure of an HTML document as a tree of objects. Each element in the HTML is represented as

a node in this tree, which can be manipulated using programming languages like JavaScript.

- **Key Features:**
  - **Access:** Use JavaScript to access and modify elements, attributes, and content in the document.
  - **Modify:** Dynamically change the structure, style, or content of a webpage.
  - **Events:** Attach event handlers to elements to make the page interactive.

What is jQuery?

- **Description:** jQuery is a fast, small, and feature-rich JavaScript library designed to simplify HTML document traversal, event handling, animation, and Ajax interactions. It allows developers to write less code to achieve more functionality, making it easier to manipulate the DOM.

Various jQuery Selectors

6. **Basic Selectors:**
   - `$("#id")`: Selects an element by ID.
   - `$(".class")`: Selects elements by class name.
   - `$("element")`: Selects all elements of a specified type (e.g., `$("p")` selects all `<p>` tags).
7. **Attribute Selectors:**
   - `$("[attribute='value']")`: Selects elements with a specific attribute value (e.g., `$("[type='text']")`).
8. **Hierarchy Selectors:**
   - `$("parent > child")`: Selects all child elements of a parent (e.g., `$("ul > li")` selects direct child `<li>` of `<ul>`).
   - `$("ancestor descendant")`: Selects all descendants of an ancestor (e.g., `$("div p")` selects all `<p>` inside `<div>`).
9. **Filter Selectors:**
   - `:first`: Selects the first matched element (e.g., `$("p:first")`).
   - `:last`: Selects the last matched element.
   - `:eq(index)`: Selects an element with a specific index (e.g., `$("li:eq(2)")` selects the third `<li>`).

These tools and concepts are fundamental to building interactive, responsive, and validated web applications.

FAQ:

1. Write 3 reasons why Form validations are important.

- A1) **Data Accuracy and Consistency:**

- Form validations ensure that the data entered by users is correct, complete, and in the expected format (e.g., valid email addresses, proper date formats), maintaining data integrity.
- **Enhanced User Experience:**
- By providing immediate feedback on errors, form validations help users correct mistakes in real-time, reducing frustration and improving overall user satisfaction.
- **Security:**
- Validating form inputs helps prevent malicious data from being submitted, which can protect against various attacks like SQL injection, XSS (Cross-Site Scripting), and other security vulnerabilities.

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

A2)

```
// Select the image element by its ID
var image = document.getElementById("myImage");

// Modify the 'src' attribute to a new value
image.setAttribute("src", "new-image.jpg");
```

In this example, the src attribute of the <img> element with the ID myImage is changed to "new-image.jpg".

3. What is jQuery Ajax?

A3) **Description:** jQuery Ajax is a set of methods provided by jQuery that allows web pages to asynchronously send and retrieve data from a server without reloading the entire page. This enables dynamic content updates and more interactive user experiences.

```
$.ajax({
  url: "example.com/api/data",
  method: "GET",
  success: function(response) {
    // Handle the successful response
    console.log(response);
  },
  error: function(error) {
    // Handle any errors
    console.error("An error occurred:", error);
  }
});
```

This example sends a GET request to a server, retrieves data, and updates the webpage dynamically based on the server's response.

Output: Screenshots of the output to be attached.

Problem Statement:

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)

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

Html:

a3 > a3.html > ...

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Student Registration Form</title>
    <link rel="stylesheet" href="a3Styles.css">
</head>
<body>
    <div class="container">

        <form id="studentTable">
            <label for="username">Username: </label>
            <input type="text" id="username" required>

            <label for="email">Email: </label>
            <input type="email" id="email" required>

            <label for="phoneNo">Phone Number: </label>
            <input type="text" id="phoneNo" required>

            <label for="password">Password: </label>
            <input type="password" id="password" required>

            <label for="confirmPass">Confirm password: </label>
            <input type="password" id="confirmPass" required>

            <button type="submit">Register</button>
        </form>
        <div id="errorMessages"></div>
```

```
                <button type="submit">Register</button>
            </form>
            <div id="errorMessages"></div>
        </div>
        <script src="a3Script.js"></script>
    </body>
    </html>
```

CSS:

a3 > # a3Styles.css > body

```css
body {
    font-family: Arial, sans-serif;
    background-color: #f4f4f4;
    margin: 0;
    padding: 0;
}

.container {
    width: 100%;
    max-width: 600px;
    margin: 50px auto;
    padding: 20px;
    background-color: #fff;
    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
    border-radius: 10px;
}

h1 {
    text-align: center;
    color: #333;
}

form {
    display: flex;
    flex-direction: column;
}

label {
    margin: 10px 0 5px;
    color: #333;
```

a3 > # a3Styles.css > body

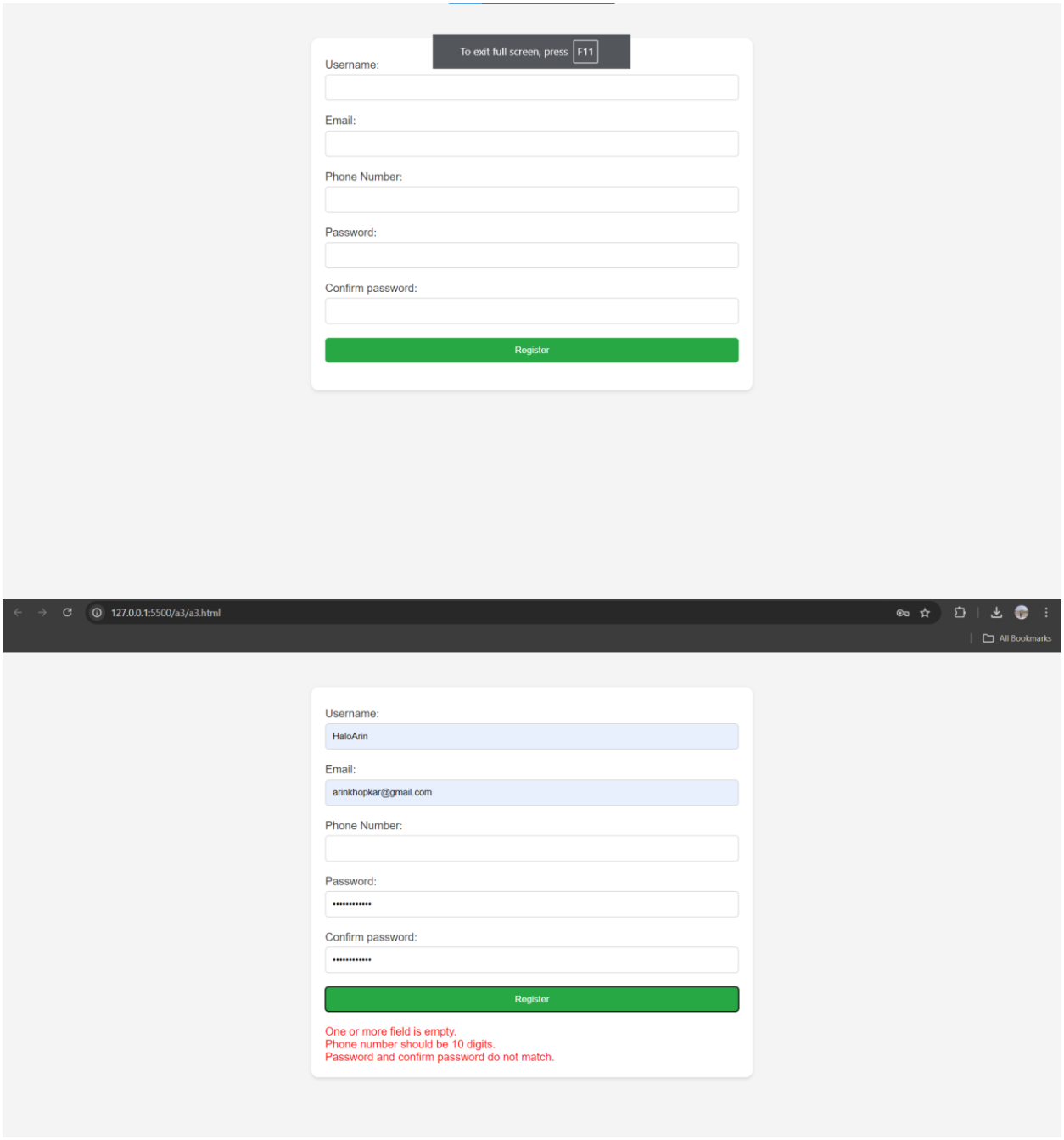
```
28    label {
31    }
32
33    input[type="text"],
34    input[type="email"],
35    input[type="password"] {
36        padding: 10px;
37        margin-bottom: 10px;
38        border: 1px solid #ccc;
39        border-radius: 5px;
40    }
41
42    button {
43        padding: 10px 20px;
44        background-color: #28a745;
45        color: white;
46        border: none;
47        border-radius: 5px;
48        cursor: pointer;
49        margin-top: 10px;
50    }
51
52    button:hover {
53        background-color: #218838;
54    }
55
56    #errorMessages {
57        color: red;
58        margin-top: 20px;
59    }
```

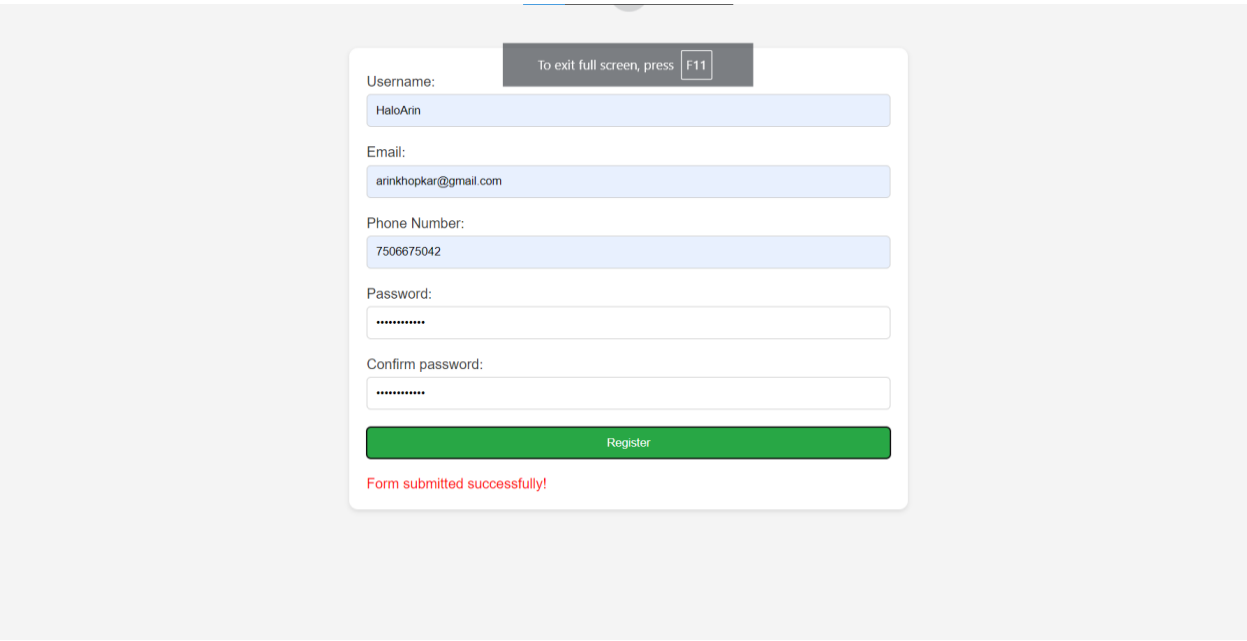
Javascript:

a3 > JS a3Script.js > ...

```javascript
document.getElementById("studentTable").addEventListener("submit", function(event)
{
    event.preventDefault();
    let errors = [];

    username = document.getElementById("username").value.trim();
    email = document.getElementById("email").value.trim();
    phoneNo = document.getElementById("phoneNo").value.trim();
    password = document.getElementById("password").value.trim();
    confirmPass = document.getElementById("confirmPass").value.trim();

    if(username === "" || email === "" || phoneNo === "" || password === "" || confirmPass === "")
    {
        errors.push('One or more field is empty.');
    }

    phonePattern = /^\d{10}$/;
    if(!phonePattern.test(phoneNo))
        errors.push("Phone number should be 10 digits.");

    emailPattern = /^[^\s@]+@[^\s@]{3,}\.[^\s@]{2,3}$/;
    if(!emailPattern.test(email))
        errors.push("Invalid email.");

    passPattern = /^(?=.*[A-Z])(?=.*\d)(?=.*[&$#@]).{7,}$/;
    if(!passPattern.test(password))
        errors.push("Password must be at least 7 characters long and contain at least one capital letter, one digit, and one special character from the set (&,$,#,@)")

    if(password !== confirmPass)
        errors.push("Password and confirm password do not match.");

    errorMessages = document.getElementById('errorMessages');
    errorMessages.innerHTML = '';
    if (errors.length > 0)
    {
        errorMessages.innerHTML = errors.join('<br>');
    }
    else
    {
        window.location.href = 'submitted.html'
        //errorMessages.innerHTML = 'Form submitted successfully!';
    }
});
```

Output form webpage:


To exit full screen, press F11
Username:
Email:
Phone Number:
Password:
Confirm password:
Register
127.0.0.1:5500/a3/a3.html
All Bookmarks
Username:
HaloArin
Email:
arinkhopkar@gmail.com
Phone Number:
Password:
Confirm password:
Register
One or more field is empty.
Phone number should be 10 digits.
Password and confirm password do not match.



DOM and jquery webpage:

Html:

a3 > a3_DOM.html > html > body > script > addEventListener('click') callback

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>DOM</title>
</head>
<body>
    <h2 id="header">Hello World</h2>

    <div id="content">
        <p class="text">This is a paragraph</p>
        <p class="text">This is another paragraph</p>
    </div>
    <p>This is just a random paragraph</p>
    <img id="image" src="/images/GitHub image.png" width="360" height="360">
    <br>

    <button id="change_text">Change the text</button>
    <button id="change_color">Change the colour</button>
    <button id="imgButton">Change Image source</button>
    <button id="addchild">Add text node</button>
    <button id="removechild">Remove child</button>
    <script>
        let childCount = 0;
        let lastChildId = null;
        document.getElementById("imgButton").addEventListener('click', function(){
            img = document.getElementById("image");
            img.src = "/images/8K.jpg";
            img.style.width = '540px';
```

a3 > a3_DOM.html > html > body > script > addEventListener('click') callback

```
2   <html lang="en">
8   <body>
24      <script>
31          });
32
33          document.getElementById("addchild").addEventListener('click', function(){
34              childCount++;
35
36              var parent = document.getElementById("content");
37
38              var child = document.createElement('div');
39              child.id = 'child-' + childCount;
40
41              var text = document.createTextNode('This is the text node ' + childCount);
42
43              child.appendChild(text);
44
45              parent.appendChild(child);
46
47              lastChildId = child.id;
48
49          });
50
51          document.getElementById("removechild").addEventListener('click', function(){
52              if(lastChildId)
53              {
54                  var lastChild = document.getElementById(lastChildId);
55
56                  lastChild.parentNode.removeChild(lastChild);
```

a3 > a3_DOM.html > html > body > script

```
<html lang="en">
<body>
    <script>
        document.getElementById("removechild").addEventListener('click', function(){

                lastChildId = (childCount > 0) ? 'child-' + childCount : null;
            }
            else
                alert('No child to delete.');
        });
        const originalParagraphTexts = [];
        document.getElementById('change_text').addEventListener('click', function(){

            const paragraphs = document.getElementsByTagName('p');

            for (let i = 0; i < paragraphs.length; i++) {
                originalParagraphTexts.push(paragraphs[i].innerHTML);
                paragraphs[i].innerHTML = `Paragraph ${i + 1}`;
            }
        });
        document.getElementById('change_color').addEventListener('click', function(){
            const textElements = document.getElementsByClassName('text');
            for (let i = 0; i < textElements.length; i++) {
                textElements[i].style.color = 'green';
            }
        });
        document.getElementById('header').innerHTML = 'This is DOM Manipulation';
        </script>
</body>
</html>
```

127.0.0.1:5500/a3/a3_DOM.html

All Bookmarks

This is DOM Manipulation

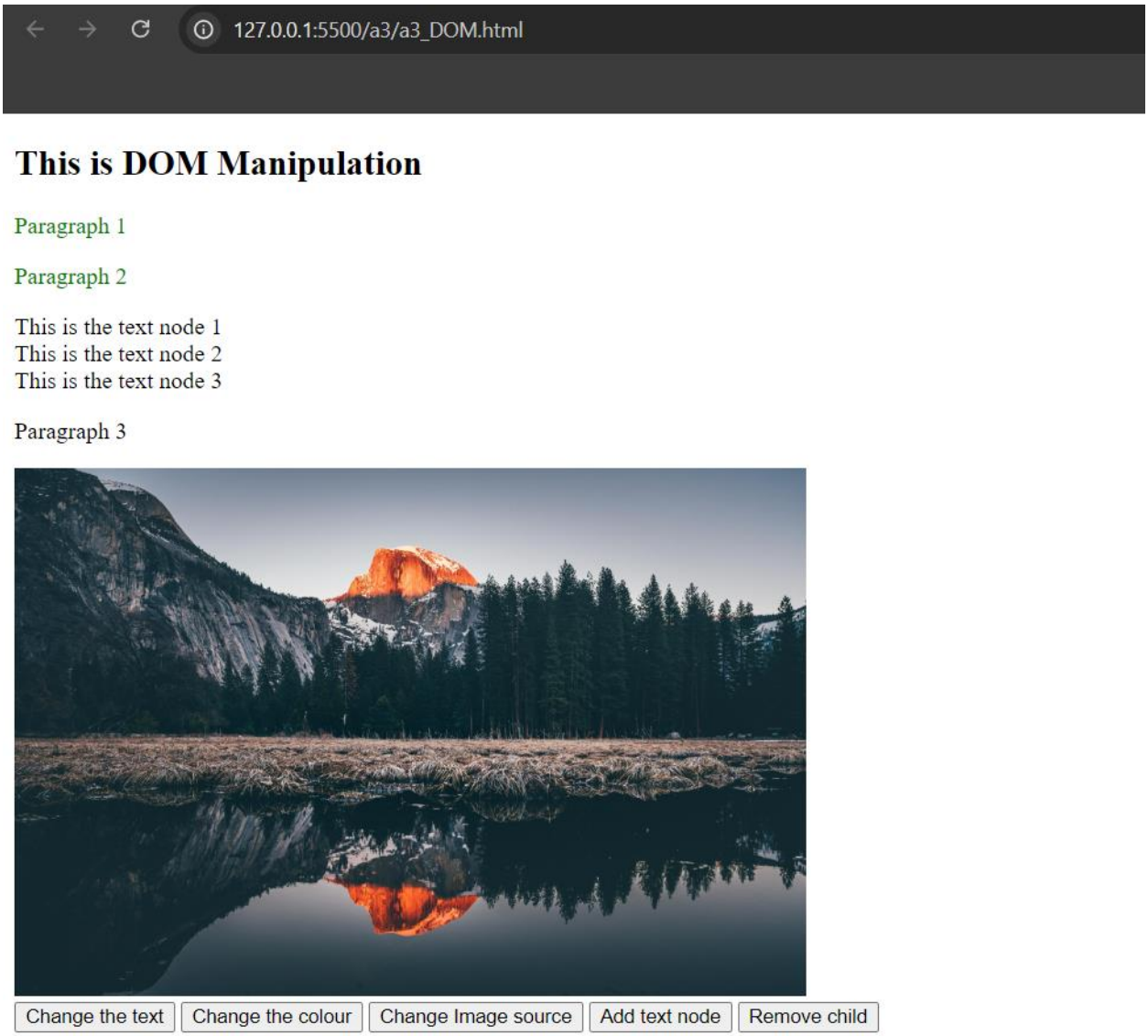
This is a paragraph

This is another paragraph

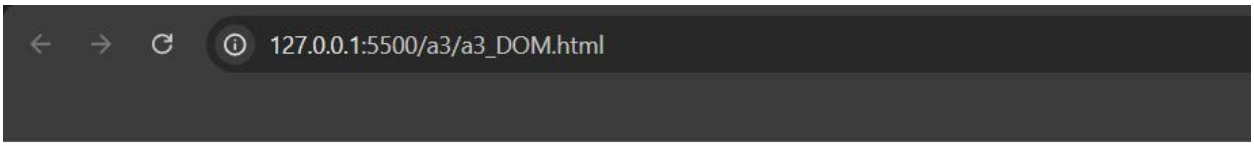
This is just a random paragraph

Change the text | Change the colour | Change Image source | Add text node | Remove child

After clicking first 4 buttons:



After Clicking last button twice:



This is DOM Manipulation

Paragraph 1

Paragraph 2

This is the text node 1

Paragraph 3

Change the text | Change the colour | Change Image source | Add text node | Remove child