Part A: DOM Fundamentals (Conceptual)

Exercise 1: Understanding DOM

Task:
• Define DOM.
The **Document Object Model (DOM)** is a programming interface for HTML documents.
It represents the webpage as a **tree structure** where each HTML element is treated as an object (node).
JavaScript uses the DOM to access, modify, create, and delete elements dynamically.

• Draw a simple DOM tree for an HTML page with <html>, <body>, <h1>, and <p>.

```
<html>

  <body>

    <h1>Title</h1>

    <p>Paragraph</p>

  </body>

</html>
```

```
html

  └──── body

      ├──── h1

      └──── p
```

Exercise 2: DOM Selection Methods

Task: Write the purpose of:
• getElementById()
Selects an element using its **id attribute**.

Eg document.getElementById("demo");

• querySelector()

Selects the **first matching element** using CSS selectors.

Eg document.querySelector(".className");

document.querySelector("#idName");
• querySelectorAll()

Selects **all matching elements** using CSS selectors.
Returns a NodeList.

Eg document.querySelectorAll("p");

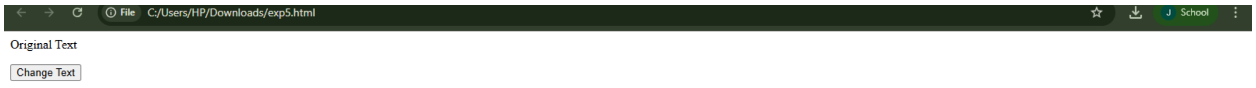
Part B: Hands-On DOM Practice

Exercise 3: Accessing Elements

Task:
• Create a paragraph with an id.
• Use JavaScript to change its text using a button click.

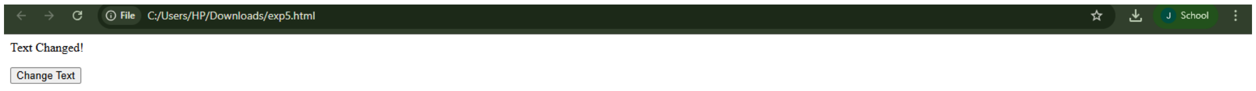
```
<p id="myPara">Original Text</p>

<button onclick="changeText()">Change Text</button>



<script>

function changeText() {

   document.getElementById("myPara").innerText = "Text Changed!";

}

</script>
```

Before



Original Text

After



Text Changed!

Change Text

Exercise 4: Creating Elements Dynamically

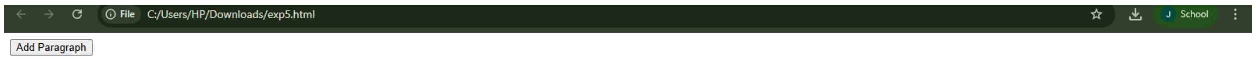
Task:

• Create a button.

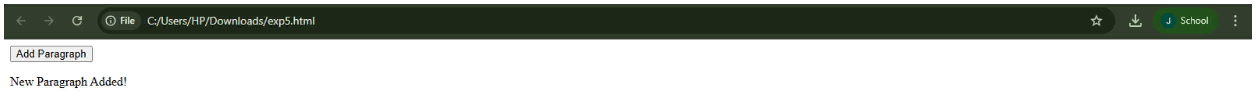
• When clicked, add a new paragraph to the page.

```
<button onclick="addParagraph()">Add Paragraph</button>



<script>

function addParagraph() {

   const newPara = document.createElement("p");

   newPara.innerText = "New Paragraph Added!";

   document.body.appendChild(newPara);

}

</script>
```

Before



After



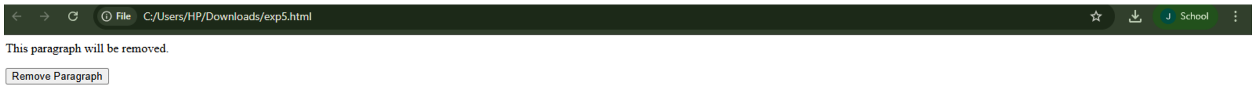
New Paragraph Added!

Exercise 5: Removing Elements

Task:
• Create a paragraph.
• Add a button to remove the paragraph using JavaScript.

```
<p id="removeMe">This paragraph will be removed.</p>

<button onclick="removeParagraph()">Remove Paragraph</button>



<script>

function removeParagraph() {

  const para = document.getElementById("removeMe");

  para.remove();

}

</script>
```

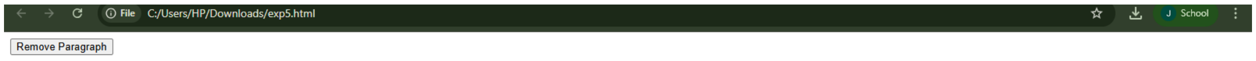
Before



C:/Users/HP/Downloads/exp5.html

This paragraph will be removed.

After



C:/Users/HP/Downloads/exp5.html

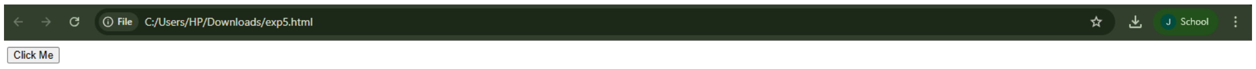
Part C: Event Handling Practice

Exercise 6: Event Listener

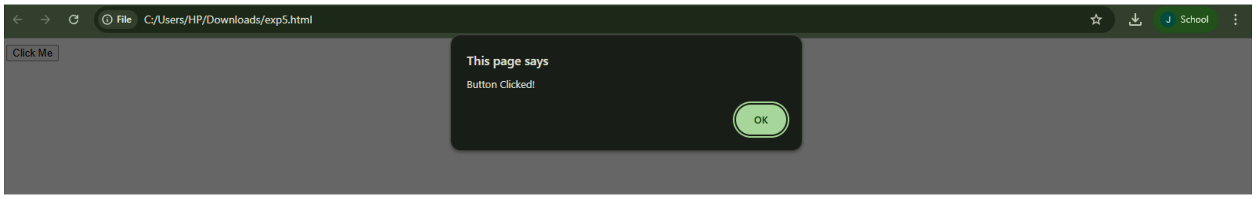
Task:
• Use addEventListener() to display a message when a button is clicked.

```
<button id="clickBtn">Click Me</button>


<script>
const btn = document.getElementById("clickBtn");


btn.addEventListener("click", function() {
   alert("Button Clicked!");
});
</script>
```

Before



After



This page says

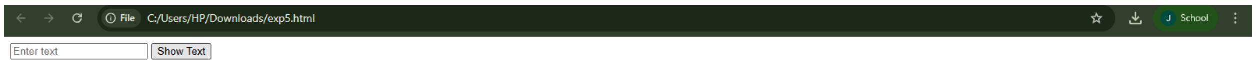
Button Clicked!

Exercise 7: Input Handling

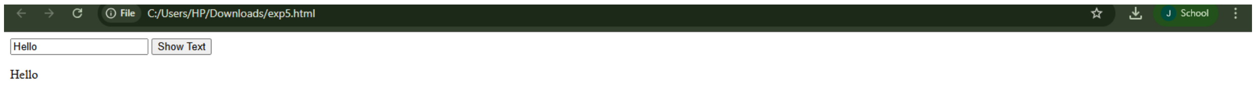
Task:
- Create a text input.
- Display the entered text below the input when a button is clicked.

```
<input type="text" id="userInput" placeholder="Enter text">

<button id="showBtn">Show Text</button>

<p id="displayText"></p>


<script>

document.getElementById("showBtn").addEventListener("click", function() {
    const text = document.getElementById("userInput").value;

    document.getElementById("displayText").innerText = text;

});

</script>
```

Before



After



Hello

Part D: Logic Preparation for To-Do List

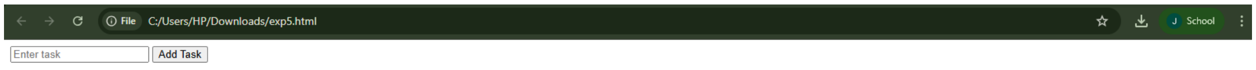
Exercise 8: Simple Task Addition Logic

Task:

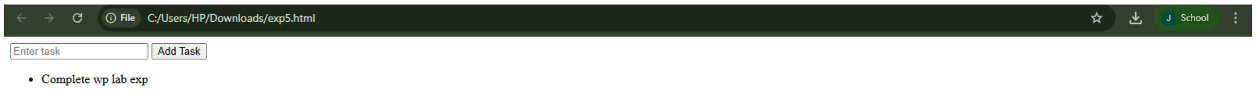
- Enter text in an input box.
- On button click, display it as a list item.

```
<input type="text" id="taskInput" placeholder="Enter task">

<button id="addTask">Add Task</button>

<ul id="taskList"></ul>



<script>

document.getElementById("addTask").addEventListener("click", function() {



    const taskValue = document.getElementById("taskInput").value;



    if (taskValue === "") {

        alert("Please enter a task");
        return;
    }

    const li = document.createElement("li");
    li.innerText = taskValue;

    document.getElementById("taskList").appendChild(li);

    document.getElementById("taskInput").value = "";
});
</script>
```

Before



After



- Complete wp lab exp