<!DOCTYPE html>

<html>

<head>

<title>DOM Manipulation with Events</title>

</head>

<body>

<h1 id="myHeading">DOM Manipulation Example</h1>

<button id="changeTextButton">Change Text</button>

<button id="changeColorButton">Change Color</button>

<script>

// Get references to the elements

const heading = document.getElementById('myHeading');

const changeTextButton = document.getElementById('changeTextButton');

const changeColorButton = document.getElementById('changeColorButton');

// Define event handlers

function changeText() {

heading.textContent = 'Text has been changed!';

}

function changeColor() {

heading.style.color = 'blue';

}

// Attach event listeners to buttons

changeTextButton.addEventListener('click', changeText);

changeColorButton.addEventListener('click', changeColor);

</script>

</body>

</html>

In this example:

We have an HTML document with a heading (<h1>), two buttons, and JavaScript code embedded within the <script> tags.

We use document.getElementById to retrieve references to the heading and buttons by their respective IDs.

We define two event handler functions (changeText and changeColor) that will be executed when the buttons are clicked.

We use addEventListener to attach click event listeners to the buttons. When the buttons are clicked, the associated event handler functions (changeText and changeColor) are executed.

The changeText function changes the text content of the heading when the "Change Text" button is clicked.

The changeColor function changes the text color of the heading when the "Change Color" button is clicked.

By using events and event listeners, you can make your web pages interactive and responsive to user actions, allowing you to manipulate the DOM in real-time to provide a dynamic user experience.