FULL STACK

EXPERIEMENT-4

**AIM**

To create a responsive and accessible web component using HTML, CSS, and JavaScript that provides a **live character counter** for a <textarea>, displaying characters used and remaining, showing visual warnings when close to or exceeding the limit, and preventing submission when the limit is exceeded.

**THEORY**

* **HTML** provides structure: <textarea> for input, elements to show counts and a submit button.
* **CSS** styles the UI and provides visual feedback (colors, spacing, responsive layout).
* **JavaScript** listens for input events, updates counters in real time, enforces a maximum length, and updates UI states (warning, error, disabled).
* **Accessibility**: use aria-live to announce count changes to screen readers and aria-describedby to link the textarea to the counter.
* **UX patterns**: show both used and remaining characters; change color when remaining gets low; disable submit when over limit.

**PROCEDURE**

1. Plan UI: textarea with label, counter area (used / remaining), progress bar, and submit button.
2. Create HTML structure with semantic elements and ARIA attributes.
3. Style with CSS for a clean responsive look and clear warning colors.
4. Add JavaScript to:
   * read maxlength (or set default),
   * update used/remaining counts on input,
   * clamp input if wanting to prevent typing beyond limit (or allow and show error),
   * update progress bar and classes for normal/warning/error states,
   * disable/enable submit accordingly, and update aria-live.
5. Test: type fast, paste long text, resize viewport, test with keyboard only and screen reader (basic).

**CODE:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Live Character Counter</title>

<style>

body {

font-family: Arial, sans-serif;

text-align: center;

padding: 50px;

background: #f4f4f4;

}

textarea {

width: 300px;

height: 120px;

padding: 10px;

font-size: 16px;

border-radius: 6px;

border: 1px solid #ccc;

resize: none;

}

.counter {

margin-top: 8px;

font-size: 14px;

color: #555;

}

</style>

</head>

<body>

<h2>Live Character Counter</h2>

<textarea id="textInput" maxlength="100" placeholder="Type here..."></textarea>

<div class="counter" id="charCounter">0 / 100</div>

<script>

const textArea = document.getElementById('textInput');

const counter = document.getElementById('charCounter');

const maxLength = textArea.maxLength;

textArea.addEventListener('input', () => {

counter.textContent = `${textArea.value.length} / ${maxLength}`;

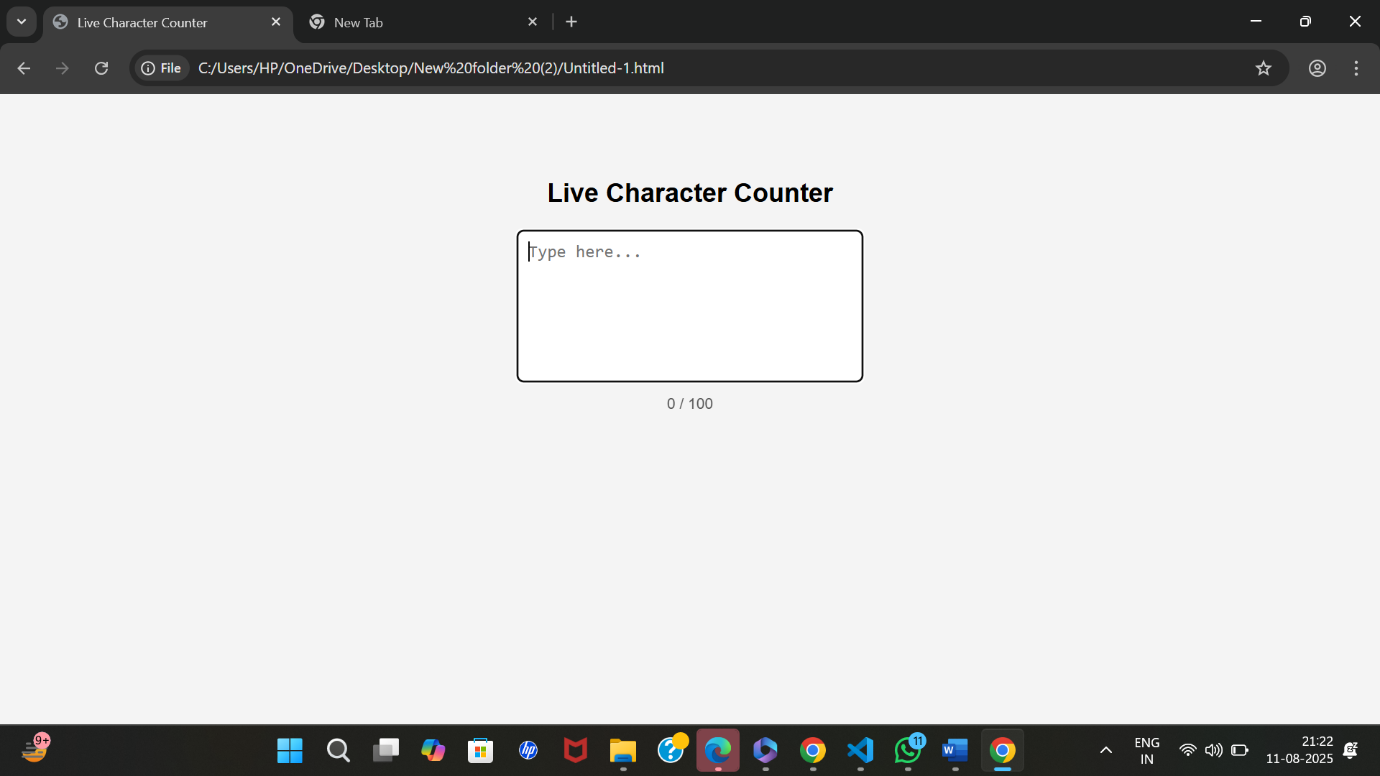
});

</script>

</body>

</html>

**OUTPUT:**

****

**Learning Outcomes:**

* Understand the use of HTML maxlength attribute.
* Learn how to capture user input with the input event in JavaScript.
* Style elements using basic CSS for better UI.