🎉 Activity: Build a Mood Button App

🧠 Objective

Create a fun web app that changes the background color and message based on the user's selected mood using **JavaScript**, **functions**, **event listeners**, and **CSS styling**.

💡 Concepts You Will Learn

- How to use `addEventListener()`
- How to pass and use **callback functions**
- The difference between **global and block scope**
- How to use **arrow functions**
- How to use **DOM manipulation** to change content and styling

---

✅ Step-by-Step Instructions

Step 1: Create the HTML File

In this step, you'll set up the structure of your web app.

📌 Instructions

1. Open VS Code (or any code editor).
2. Create a new file and save it as `index.html`.
3. Add the basic structure of an HTML page:
    - Start with `<!DOCTYPE html>`.
    - Add `<html>`, `<head>`, and `<body>` tags.
4. Inside the `<head>` section:
    - Add a `<title>` (e.g., "Mood Button App").
    - Link a CSS file named `style.css`.
5. Inside the `<body>` section:
    - Add a heading: `<h1>What's Your Mood?</h1>`
    - Create four `<button>` elements for different moods.
    - Add a `<div id="message">` to display a custom message.
    - Link your JavaScript file `script.js` at the bottom using `<script src="script.js"></script>`.

Step 2: Style the Mood App (CSS)

You'll now add styling to make your app look fun and visually appealing.

📌 Instructions

1. Create a new file and save it as `style.css`.
2. Set the background color and font for the page.
3. Style the heading and message box.
4. Make the buttons large, colorful, and fun to click.
5. Create **custom classes** for different moods (happy, sad, angry, surprised) with different background colors.

💡 Hints

- Use `transition` to smoothly change background color.
- Use `border-radius` and `hover` effects to make buttons more interactive.
- Use `.happy`, `.sad`, `.angry`, and `.surprised` classes for color themes.

Step 3: Set Up JavaScript File

You'll now write the code to make the buttons change the mood and message.

📌 Instructions

1. Create a new file called `script.js`.
2. Get references to the `<body>` and `#message` elements using `document.body` and `document.getElementById()`.
3. Define a function called `setMood(moodName, moodClass, callback)` that:
    - Clears previous mood classes
    - Adds the new mood class to the body
    - Calls a callback function to update the message
4. Define a `showMessage(mood)` function to update the text content inside `#message`.

Step 4: Add Event Listeners

Now you'll make the buttons interactive.

📌 Instructions

1. Add `click` event listeners to each mood button using `addEventListener`.
2. Inside each listener, call `setMood()` with:
    - The mood name (e.g., "happy")
    - The matching CSS class (e.g., "happy")
    - A callback function (like `showMessage`) to update the message

💡 Hints

- Try using an **arrow function** as a callback for at least one button!
- You can use an **inline anonymous function** to customize the message even more.

Step 5: Clear Previous Mood (Helper Function)

Create a helper function called `clearMoods()` to remove all mood-related classes from the `<body>`.

📌 Instructions

- Use `classList.remove("happy", "sad", "angry", "surprised")`.

Step 6: Test Your Mood App

Make sure everything works as expected!

📌 Instructions

1. Open `index.html` in a browser.
2. Click each button and observe:
    - The background color changes.
    - The message updates based on the mood.
3. Check your console for any errors if it’s not working.

There is a tutorial video for more help if needed:

https://www.youtube.com/watch?v=uFkaHzUHWRg