

# DAY 14

## Index.html

### <body> – The main content

#### 1. Heading

```
<h1>Daily Mental Health Tracker</h1>
```

Shows the title of the app.

#### 2. Inputs Section

```
<input type="number" id="water">
```

```
<input type="number" id="exercise">
```

```
<input type="text" id="mood" readonly>
```

- Water input – how many cups you drank
- Exercise input – how many minutes you exercised
- Mood – automatically updates based on the other two inputs.  
(You can't type in the mood box – it's filled in automatically)

#### 3. Buttons

```
<button id="submit">Add Entry</button>
```

```
<button id="updateEntry">Update Entry</button>
```

```
<button id="cancelEdit">Cancel</button>
```

- **Add Entry:** Adds your input to the table.
- **Update Entry:** Changes a saved row (only shows when editing).
- **Cancel:** Cancels editing and clears inputs.

#### 4. Table for Results

```
<table>
```

```
<thead>...</thead>
```

```
<tbody id="entryTable">...</tbody>
```

```
</table>
```

- Shows your **history of entries** with water, exercise, mood, and buttons to edit/delete each one.

## Style.css

```
body {
```

```
margin: 0;
```

```
font-family: Arial, sans-serif;
```

```
background: url('red-reyes-lvMR5ebekPM-unsplash.jpg') no-repeat center center fixed;
```

```
background-size: cover;
```

```
padding: 20px;
```

}

- Removes all outer space using `margin: 0`.
- Sets the text font to Arial.
- Adds a background image (a gym image).
- Makes the image **fixed** (it stays still when scrolling).
- Covers the entire screen (`background-size: cover`).
- Adds space inside the page with `padding: 20px`.

#### **.app**

```
.app {  
  max-width: 700px;  
  margin: 50px auto;  
  background: rgba(255, 255, 255, 0.85);  
  backdrop-filter: blur(5px);  
  padding: 30px;  
  border-radius: 12px;  
  box-shadow: 0 8px 16px rgba(0, 0, 0, 0.3);  
}
```

- Makes the main box only **700px wide**.
- Centers it horizontally with `margin: 50px auto`.
- Gives it a **semi-transparent white** background.
- Adds a **blur effect** behind the box.
- Adds **rounded corners** and **inner padding**.
- Adds a **soft shadow** to make it pop.

#### **h1 (Page Heading)**

```
h1 {  
  text-align: center;  
  color: #222;  
  margin-bottom: 25px;  
}
```

- Centers the title text.
- Makes it dark gray (`#222`).
- Adds space below it.

#### **.inputs (All input sections)**

```
.inputs {  
  display: flex;  
  justify-content: space-between;  
  gap: 20px;  
  flex-wrap: wrap;  
  margin-bottom: 20px;  
}
```

- Uses a **flex layout** to arrange inputs side by side.

- Adds gaps between them.
- Allows wrapping on small screens (flex-wrap: wrap).
- Adds space below the inputs.

#### **.inputs div**

```
.inputs div {
  flex: 1;
  min-width: 150px;
}
```

- Each input block will **take equal space**.
- But won't go smaller than 150px.

#### **label**

```
label {
  display: block;
  margin-bottom: 5px;
  font-weight: bold;
}
```

- Each label appears on a **new line**.
- Adds space below each label.
- Makes the text bold.

#### **input**

```
input {
  width: 100%;
  padding: 8px;
  border-radius: 6px;
  border: 1px solid #ccc;
}
```

- Makes the input **fill its full width**.
- Adds padding inside the input box.
- Gives it **rounded corners**.
- Adds a light gray border.

#### **.buttons**

```
.buttons {
  text-align: center;
  margin-bottom: 20px;
}
```

- Centers the buttons.
- Adds space below the button section.

#### **.buttons button**

```
.buttons button {
padding: 10px 20px;
margin: 0 5px;
background-color: #4caf50;
color: white;
border: none;
border-radius: 6px;
cursor: pointer;
}
```

- Adds padding inside buttons.
- Adds space between buttons (margin).
- Green background (#4caf50).
- White text, no border.
- Rounded edges.
- Makes it clickable (cursor: pointer).

#### **.buttons button:hover**

```
.buttons button:hover {
background-color: #45a049;
}
```

- Changes button color slightly when you hover.

#### **table**

```
table {
width: 100%;
border-collapse: collapse;
background-color: white;
border-radius: 6px;
overflow: hidden;
}
```

- Table takes full width.
- Removes spacing between table borders (border-collapse: collapse).
- White background.
- Rounded corners.
- Hides anything going outside the border.

```
th, td {
padding: 12px;
border-bottom: 1px solid #ddd;
text-align: center;
}
```

- Adds padding to table headers and cells.
- Adds light gray lines under rows.
- Centers the text.

```
th {
```

```
background-color: #4caf50;
color: white;
}
```

- Makes header row green with white text.

#### **td button**

```
td button {
  background-color: #f44336;
  color: white;
  border: none;
  padding: 6px 12px;
  cursor: pointer;
  border-radius: 4px;
}
```

- Red buttons inside the table (for delete/edit).
- White text, no border, rounded.
- Clickable.

#### **td button:hover**

```
td button:hover {
  background-color: #d32f2f;
}
```

- Darker red on hover.

## **Script.js**

### **1. Captures the Elements on the Page**

```
const waterInput = document.getElementById('water');
const exerciseInput = document.getElementById('exercise');
const moodInput = document.getElementById('mood');
```

- These lines **connect** your code to the boxes (inputs) where the user types:
  - Water intake
  - Exercise time
  - Mood (calculated automatically)

### **2. Stores Your Entries**

```
let entries = [];
let editingIndex = null;
```

- `entries[]` is an **empty list** that will store each day's data.
- `editingIndex` remembers **which row you're editing**.

### 3. Auto-Calculate Mood Based on Water & Exercise

```
function calculateMood() { ... }
```

- If someone drank **8+ glasses** and exercised **30+ minutes**, mood = ☺ Happy.
- If 5+ glasses and 15+ minutes, mood = 😐 Neutral.
- Less than that, mood = ☹ Sad.
- It updates automatically when user types values.

### 4. Run Mood Calculation When User Types

```
waterInput.addEventListener('input', calculateMood);  
exerciselnput.addEventListener('input', calculateMood);
```

- This means: Whenever the user types in **water** or **exercise**, the **mood gets updated**.

### 5. Add a New Entry

```
submitBtn.addEventListener('click', () => { ... });
```

- When the user clicks **Submit**, it:
  - Takes today's date.
  - Collects water, exercise, and mood values.
  - Saves it in the entries[] list.
  - Shows the new entry in the table.
  - Clears the form for the next entry.

### 6. Display Entries in the Table

```
function renderEntries() { ... }
```

- This function shows all saved entries in the **table**.
- It adds a row for each entry with **Edit** and **Delete** buttons.

### 7. Edit an Existing Entry

```
window.editEntry = function(index) { ... }
```

- When you click **Edit**:
  - It fills the form with that entry's values.
  - Hides the Submit button and shows **Update** and **Cancel**.
  - Remembers which entry you're editing.

### 8. Update an Edited Entry

```
updateBtn.addEventListener('click', () => { ... });
```

- When you click **Update**:

- It **replaces the old data** with the new form data.
- Updates the date to today again.
- Shows the changes in the table.
- Clears the form.

## 9. Cancel Editing

```
cancelBtn.addEventListener('click', resetForm);
```

- Clicking **Cancel** just clears the form and exits editing mode.

## 10. Delete an Entry

```
window.deleteEntry = function(index) { ... }
```

- When you click **Delete**, it removes that row from the list and refreshes the table.

## 11. Reset the Form

```
function resetForm() { ... }
```

- Clears all input boxes.
- Hides Update and Cancel buttons.
- Shows Submit button again.