

Project:1-Expense Tracker

Name: Sri Nithyasri R(717823T153)

Department: Electronics and TeleCommunication Engineering

Introduction:

An **Expense Tracker** is a web application that helps users record, manage, and analyze their financial expenditures. Using **JavaScript**, we can create an interactive and dynamic tracker with features like adding expenses, categorizing them, viewing summaries, and analyzing spending trends.

Key Components:

1. **Frontend:** HTML for structure, CSS for styling, and JavaScript for interactivity.
2. **Data Storage:** LocalStorage for saving expense data.
3. **Features:**
 - Input fields for expense details (amount, category, date, etc.).
 - Dynamic table or list to display expenses.
 - Graphs (using libraries like Chart.js) for visualizing data.
 - Filtering and sorting options for better analysis.
4. **Tools:** JavaScript frameworks (optional) like React.js

This application enhances financial awareness and can be a foundation for learning JavaScript development.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Sri's official website for tracking expenses</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
    }
  </style>
</head>
<body>
  <div>
    <h1>Expense Tracker</h1>
    <div>
      <input type="text" value="Amount" />
      <input type="text" value="Category" />
      <input type="text" value="Date" />
      <button>Add Expense</button>
    </div>
    <table>
      <thead>
        <tr>
          <th>Amount</th>
          <th>Category</th>
          <th>Date</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <td><input type="text" value="Amount" /></td>
          <td><input type="text" value="Category" /></td>
          <td><input type="text" value="Date" /></td>
        </tr>
      </tbody>
    </table>
  </div>
</body>
</html>
```

```
.main-container {
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  height: 100vh;
  text-align: center;
}

.login-container {
  display: flex;
  flex-direction: column;
  gap: 20px;
  padding: 30px;
  border: 1px solid #ccc;
  border-radius: 10px;
  width: 300px;
  background-color: #f9f9f9;
}

.login-container input {
  padding: 10px;
  font-size: 16px;
  border-radius: 5px;
  border: 1px solid #ccc;
}

.login-container button {
  padding: 10px 20px;
  font-size: 16px;
  cursor: pointer;
  background-color: #4CAF50;
  color: white;
  border: none;
  border-radius: 5px;
}

.login-container button:hover {
  background-color: #45a049;
}

.header-container {
  display: flex;
  align-items: center;
  gap: 10px;
  margin-bottom: 20px;
}

.header-container img {
  width: 50px;
  height: 50px;
  object-fit: cover;
  border-radius: 50%;
}
```

```
.header-container h1 {
  font-size: 24px;
  margin: 0;
}
button {
  padding: 10px 20px;
  font-size: 16px;
  cursor: pointer;
  border: none;
  border-radius: 5px;
  background-color: #4CAF50;
  color: white;
}
button:hover {
  background-color: #45a049;
}
.expense-form {
  display: flex;
  flex-direction: column;
  gap: 10px;
  margin-top: 20px;
}
.expense-form input {
  padding: 10px;
  font-size: 16px;
  border-radius: 5px;
  border: 1px solid #ccc;
}
.expense-list {
  margin-top: 20px;
  width: 80%;
  max-width: 600px;
  text-align: left;
}
.expense-list ul {
  list-style-type: none;
  padding: 0;
}
.expense-list li {
  display: flex;
  justify-content: space-between;
  align-items: center;
  background-color: #f9f9f9;
  margin-bottom: 5px;
  padding: 10px;
  border-radius: 5px;
  flex-wrap: wrap;
}
```

```
.expense-list li span {
  flex: 1;
  padding-right: 10px;
}
.expense-list li button {
  background-color: #f44336;
  padding: 5px 10px;
  border-radius: 5px;
}
.expense-list li button:hover {
  background-color: #d32f2f;
}
@media (max-width: 600px) {
  .expense-list li {
    flex-direction: column;
    align-items: flex-start;
  }
  .expense-list li span {
    width: 100%;
    margin-bottom: 10px;
  }
  .expense-list li button {
    width: 100%;
    text-align: center;
  }
}
.total-amount {
  font-size: 18px;
  font-weight: bold;
  margin-top: 20px;
}
.modal {
  display: none;
  position: fixed;
  z-index: 1;
  left: 0;
  top: 0;
  width: 100%;
  height: 100%;
  background-color: rgba(0, 0, 0, 0.4);
}
.modal-content {
  background-color: white;
  padding: 20px;
  margin: 15% auto;
  width: 300px;
  text-align: center;
  border-radius: 10px;
}
```

```

    }
    .modal button {
        margin-top: 10px;
    }
</style>
</head>
<body>
    <div id="loginPage" class="main-container">
        <div class="login-container">
            <h2>Login</h2>
            <input type="password" id="password" placeholder="Enter password"
required>
            <button onclick="login()">Log In</button>
            <p id="errorMessage" style="color: red; display: none;">Incorrect
password. Please try again.</p>
        </div>
    </div>
    <div id="mainContent" class="main-container" style="display:none;">
        <div class="header-container">
            
            <h1>Welcome to Sri's official website for tracking your
expenses</h1>
        </div>
        <button onclick="startTracker()">Start tracking your EXPENSES</button>
        <div class="expense-form" id="expenseForm" style="display:none;">
            <input type="text" id="description" placeholder="Enter Expense
description" required>
            <input type="number" id="amount" placeholder="Enter the amount"
required>
            <button onclick="addExpense()">Add Expense</button>
        </div>
        <div class="expense-list" id="expenseList" style="display:none;">
            <h3>Expense Details</h3>
            <ul id="expenseDetails"></ul>
        </div>
        <div class="total-amount" id="totalAmount" style="display:none;">
            Total: Rs. <span id="total"></span> /-
        </div>
    </div>
    <div id="confirmationModal" class="modal">
        <div class="modal-content">
            <p>Are you sure you want to remove this expense?</p>
            <button id="confirmDeleteBtn">Yes</button>
            <button id="cancelDeleteBtn">No</button>
        </div>
    </div>

```

```

</div>

<script>
  function login() {
    const password = document.getElementById("password").value;
    const correctPassword = "sri12345";
    const errorMessage = document.getElementById("errorMessage");
    if (password === correctPassword) {
      document.getElementById("loginPage").style.display = "none";
      document.getElementById("mainContent").style.display = "flex";
    } else {
      errorMessage.style.display = "block";
    }
  }

  class ExpenseTracker {
    constructor() {
      this.expenses = [];
      this.total = 0;
    }
    addExpense(description, amount) {
      const expense = {
        description: description,
        amount: parseFloat(amount),
        date: new Date().toLocaleString()
      };
      this.expenses.push(expense);
      this.total += expense.amount;
      this.displayExpenses();
    }
    removeExpense(index) {
      document.getElementById('confirmationModal').style.display =
'block';

      document.getElementById('confirmDeleteBtn').onclick = () => {
        const removedExpense = this.expenses.splice(index, 1)[0];
        this.total -= removedExpense.amount;
        this.displayExpenses();
        document.getElementById('confirmationModal').style.display
= 'none';
      };
      document.getElementById('cancelDeleteBtn').onclick = () => {
        document.getElementById('confirmationModal').style.display
= 'none';
      };
    }
    displayExpenses() {
      const expenseDetailsElement =
document.getElementById("expenseDetails");
      expenseDetailsElement.innerHTML = '';

```

```

        this.expenses.forEach((expense, index) => {
            const li = document.createElement("li");
            li.innerHTML = `<span>${expense.description} -
Rs. ${expense.amount.toFixed(2)} /- (Date: ${expense.date})</span>
<button
onclick="removeExpense(${index})">Remove</button>`;
            expenseDetailsElement.appendChild(li);
        });
        document.getElementById("total").textContent =
this.total.toFixed(2);
    }
}
let myTracker;
function startTracker() {
    myTracker = new ExpenseTracker();
    document.getElementById("expenseForm").style.display = "block";
    document.getElementById("expenseList").style.display = "block";
    document.getElementById("totalAmount").style.display = "block";
}
function addExpense() {
    const description = document.getElementById("description").value;
    const amount = document.getElementById("amount").value;

    if (description && amount && !isNaN(amount)) {
        myTracker.addExpense(description, amount);
        document.getElementById("description").value = '';
        document.getElementById("amount").value = '';
    }
}
function removeExpense(index) {
    myTracker.removeExpense(index);
}
</script>
</body>
</html>

```

Gist of the code:

(Designing the Expense Tracker Code)

The code is thoughtfully designed to provide a smooth and user-friendly experience using **HTML**, **CSS**, and **JavaScript**.

1. HTML Structure

The HTML layout is straightforward and intuitive:

- The **login page** has a simple input for the password and a login button.

- The **main page** includes a welcoming header with an image, a button to start expense tracking, a form for entering expenses, and sections for displaying expense details and the total amount.
- A **confirmation modal** pops up when you try to delete an expense, making sure you really want to remove it.

2. CSS Styling

The design is clean and modern:

- **Centered layouts** using Flexbox ensure everything is neat and easy to focus on.
- **Reusable styles** like rounded buttons, consistent padding, and hover effects make interactions smooth and visually appealing.
- **Color-coded actions** (green for adding and red for deleting) make the interface intuitive.
- The site is **responsive**, adapting to smaller screens with a mobile-friendly layout.

3. JavaScript Functionality

The interactive features are powered by JavaScript:

- **Login handling** makes sure only authorized users can access the tracker.
- An **ExpenseTracker class** handles all the data, including adding, removing, and displaying expenses. It also updates the total amount automatically.
- A **confirmation modal** ensures you don't accidentally delete an expense.
- Everything updates dynamically—when you add or remove an expense, the page reflects the changes instantly.

4. User-Friendly Features

The design is made with users in mind:

- Clear error messages for incorrect passwords guide users back on track.
- Buttons and input fields are visually engaging with hover effects and padding.
- The layout is simple and uncluttered, making it easy to navigate.

5. Mobile-Friendly Design

The site works just as well on mobile devices as it does on desktops, thanks to responsive design techniques. Everything adjusts beautifully, so you can track your expenses on the go.

Overall, the code is designed to be simple yet effective, ensuring anyone can use the tracker without hassle.

Output:

Login page

Login

Log In

Main page



Welcome to Sri's official website for tracking your expenses

Start tracking your EXPENSES

Overview



Welcome to Sri's official website for tracking your expenses

Start tracking your EXPENSES

Enter Expense description

Enter the amount

Add Expense

Expense Details

Total: Rs. /-

Add Expense



Welcome to Sri's official website for tracking your expenses

Start tracking your EXPENSES

Tea

320

Add Expense

Expense Details

Coffee - Rs.240.00 /- (Date: 28/12/2024, 2:18:30 pm)

Remove

Total: Rs. 240.00 /-

Updation in total amount spent on expenses



Welcome to Sri's official website for tracking your expenses

Start tracking your EXPENSES

Enter Expense description

Enter the amount

Add Expense

Expense Details

Coffee - Rs.240.00 /- (Date: 28/12/2024, 2:18:30 pm)

Remove

Tea - Rs.320.00 /- (Date: 28/12/2024, 2:20:11 pm)


Remove

Chocolate - Rs.4000.00 /- (Date: 28/12/2024, 2:20:33 pm)

Remove

Total: Rs. 4560.00 /-

Alert for removing an expense from the list

**Welcome to Sri's official website for tracking your expenses**

Enter Expense

Add Expense

Are you sure you want to remove this expense?

Yes


No

Expense Details

Coffee - Rs.240.00 /- (Date: 28/12/2024, 2:18:30 pm)	Remove
Tea - Rs.320.00 /- (Date: 28/12/2024, 2:20:11 pm)	Remove
Chocolate - Rs.4000.00 /- (Date: 28/12/2024, 2:20:33 pm)	Remove

Total: Rs. 4560.00 /-

Removal of the selected element from the list

**Welcome to Sri's official website for tracking your expenses**

Start tracking your EXPENSES

Enter Expense description

Enter the amount

Add Expense

Expense Details

Tea - Rs.320.00 /- (Date: 28/12/2024, 2:20:11 pm)	Remove
Chocolate - Rs.4000.00 /- (Date: 28/12/2024, 2:20:33 pm)	Remove

Total: Rs. 4320.00 /-

(Coffee got removed from the list)