# Expense Tracker Project — Step-by-Step Guide for Beginners

### Step 1: Define the purpose (MVP)

* Add expense (title + amount)
* Show all expenses in a list
* Delete an expense
* Show total of all expenses
* Persist expenses (optional with localStorage)

---

### Step 2: Plan components

* `App.jsx`  Holds the `expenses` state
* `ExpenseForm.jsx`  Input form to add expense
* `ExpenseList.jsx`  List all expenses
* `ExpenseItem.jsx`  Single expense with delete button

---

### Step 3: Decide data structure

```js
{
id: 123, // unique identifier
title: "Tea",
amount: 20.5
}
```

---

### Step 4: Decide state & data flow

* State lives in App
* Child ▯ parent communication via props functions:

* `onAddExpense(newExpense)`
* `onDelete(id)`

---

### Step 5: Build one feature at a time

1. Form ▯ add expense
2. List ▯ display expenses
3. Delete ▯ remove expense by id
4. Total ▯ sum of all expenses
5. Persist ▯ localStorage (optional)

---

### Step 6: Test after each feature

* Test add ▯ list ▯ delete ▯ total
* Debug using `console.log`

---

### Step 7: Refactor

* Keep components small
* Remove repeated code
* Use `id` instead of index

---

### Step 8: Deploy & polish

* Add basic CSS

* Deploy to Vercel/Netlify
* Write README with: Features / How to run / Tech used / What I learned

---

### Step 9: Optional improvements

* Edit expenses
* Filter/sort
* Validate input
* Add categories or dates

---

### Key Tips

* Break the app into tiny pieces
* Implement one feature at a time
* Always log state & props to debug
* Commit changes regularly on GitHub

---

**Outcome:** By following this guide, you will have a **fully working Expense Tracker** you built yourself without watching a tutorial.