AI In Development: Practice

## **1. Web Application: Expense Calculator**

**Goal:**

Develop a web application to calculate the main indicators of monthly expenses based on a user's list of expenses:

* Total amount of expenses
* Average daily expense
* Top 3 largest expenses

**Tools:**

* CursorAI for generating HTML/JS code
* CodePen / JSFiddle for testing

**Input Format:**

The user enters a list of their expenses in the form of a table:

| **Category** | **Amount ($)** |
| --- | --- |
| Groceries | 15,000 |
| Rent | 40,000 |
| Transportation | 5,000 |
| Entertainment | 10,000 |
| Communication | 2,000 |
| Gym | 3,000 |

**Functionality:**

The application should provide the user with the following features:

1. Adding new expenses to the list
2. Calculating the total amount of expenses (for example, for the data given above: 75,000 $)
3. Calculating the average daily expense (75,000 / 30 ≈ 2,500 $)
4. Displaying the top 3 largest expenses (Rent (40,000), Groceries (15,000), Entertainment (10,000))

**Expected Result:**

After entering data and clicking the "Calculate" button, the following should be displayed on the screen:

* Total amount of expenses
* Average daily expense
* Top 3 expenses

## **2. API Testing: Identifying Defects in Product Data**

**Task:**

Develop automated tests to validate data provided by a public API to detect errors and anomalies.

**Tools:**

* CursorAI for generating test scenarios or ChatGPT.
* ReqBin (reqbin.com) or Postman for executing API requests.
* API:<https://fakestoreapi.com/products> (mock store).

**Initial Data:**

A GET request to<https://fakestoreapi.com/products> returns an array of objects representing products. The provided JSON data contains defects that need to be identified.

**Test Objectives:**

* Verify server response code (expected 200).
* Confirm the presence of the following attributes for each product:
  + `title` (name) - must not be empty.
  + `price` (price) - must not be negative.
  + `rating.rate` - must not exceed 5.
* Generate a list of products containing defects.

## **3. SQL Queries: Analyzing a Database Online**

**Goal:** Write SQL queries to analyze sales data for an online store.

**Tools to Use:**

* [SQLite Online](https://sqliteonline.com/)
* CursorAI/ChatGPT to generate and refine SQL queries

### **Input Data (Script to Populate the Table):**

Run this script in **SQLite Online**:

CREATE TABLE orders (

id INTEGER PRIMARY KEY,

customer TEXT,

amount REAL,

order\_date DATE

);

INSERT INTO orders (customer, amount, order\_date) VALUES

('Alice', 5000, '2024-03-01'),

('Bob', 8000, '2024-03-05'),

('Alice', 3000, '2024-03-15'),

('Charlie', 7000, '2024-02-20'),

('Alice', 10000, '2024-02-28'),

('Bob', 4000, '2024-02-10'),

('Charlie', 9000, '2024-03-22'),

('Alice', 2000, '2024-03-30');

### **Tasks:**

1. Calculate the total sales volume for March 2024.
2. Find the customer who spent the most overall.
3. Calculate the average order value for the last three months.

### **Expected Results:**

1. Total sales for March: 27,000
2. Top-spending customer: Alice (20,000 )
3. Average order value (total sales / number of orders): 6,000