Junior Developer Home Exercise: Social Media Posts Management System

Overview

Build a full-stack application to manage and display social media posts data from a corrupted CSV file.

Estimated Time: 6-8 hours
Deadline: Submit within **72 hours** of receiving this task
Submission: ZIP file containing source code, README, and demo video

---

CSV Data Structure

You are provided with a CSV file (social_media_posts_data.csv) containing the following columns:

Column Name	Data Type	Description
post_id	INTEGER	Unique identifier for the post
author_first_name	TEXT	Author's first name
author_last_name	TEXT	Author's last name
author_email	TEXT	Author's email address
author_company	TEXT	Author's company name
author_job_title	TEXT	Author's job title
author_bio	TEXT	Author's biography
author_follower_count	INTEGER	Number of followers
author_verified	BOOLEAN	Whether author is verified
post_text	TEXT	Content of the post

post_date	DATETIME	When the post was created
likes	INTEGER	Number of likes
comments	INTEGER	Number of comments
shares	INTEGER	Number of shares
total_engagements	INTEGER	Total engagements (likes + comments)
engagement_rate	DECIMAL	Engagement rate percentage
post_image_svg	TEXT	SVG image code for the post
post_category	TEXT	Category of the post
post_tags	TEXT	JSON array of hashtags
location	TEXT	Post location

Part 1: Data Processing & Database Design (40%)

Task

Analyze the CSV file, identify and fix data quality issues, then import into a SQLite database.

Requirements

1. **Data Analysis & Cleaning**
    - Identify data quality issues in the CSV
    - Fix inconsistent data types, formats, and corrupted values
    - Document all issues found and solutions
2. **Database Design**
    - Design a normalized SQLite database schema
    - The CSV has denormalized data - structure it properly
    - Write migration scripts or schema creation code
3. **Data Import**

- Your application should automatically create the database schema and import the data
- The code should handle the data cleaning and import process

---

Part 2: Backend API Development (35%)

Task

Create a backend API that supports all the functionality shown in the UI mockup.

Requirements

- Choose your preferred language: **JavaScript/TypeScript, Java, or Python**
- Design API endpoints based on what the UI needs
- Your code should create the SQLite database schema automatically

---

Part 3: Frontend Development (25%)

Task

Build a React UI matching the provided design figma .

Requirements

- Use React
- Implement all features shown in the mockup

Submission Requirements

1. README.md:

- Setup instructions (step-by-step)
- How to run the application
- Environment variables needed
- Data issues you found and how you fixed them

2. Demo Video Requirements:

- **Maximum 5 minutes** screen recording
- Show the application running
- Demonstrate all features:
  - Viewing posts
  - Filtering and sorting

    - Search functionality
    - Adding a new post
    - Deleting a post

3. Code Requirements:

- Your application should **automatically create the database schema** when run
- The app should run by following your README instructions

Deadline

Submit within 72 hours of receiving this task.

Good luck!