WindBorne Junior Finance Automation Engineer Take-Home Assignment

Objectives: Create a Python or JavaScript application that will query data from Alpha Vantage API, analyze key financial metrics, and design a system to automate and sync this data.

Documentation: https://www.alphavantage.co/documentation/

Part 1: Data Extraction
Fetch financial statements for these 3 public companies using Alpha Vantage API:

- TE Connectivity — `TEL`
- Sensata Technologies — `ST`
- DuPont de Nemours — `DD`

Pull last 3 years of data for each:

- Income Statement (annual)
- Balance Sheet (annual)
- Cash Flow Statement (annual)

Part 2: Data Transformation
Design and populate a PostgreSQL schema with:

- companies table (company metadata)
- financial_statements table (normalized financial data)

Should handle multiple statement types and periods
Key metrics should be queryable
Transform the API responses into your schema.

Part 3: Financial Analysis & Visualizations
Pick 3-5 financial metrics, calculate and store these metrics for each company by year. Please use either Python or Javascript.

Example metrics:

- Profitability: Gross Margin %, Operating Margin %, Net Margin %
- Liquidity: Current Ratio, Quick Ratio
- Efficiency: Asset Turnover, Days Sales Outstanding (if calculable)
- Growth: Revenue YoY %, Net Income YoY %

Use your creativity to create a simple dashboard to display these metrics:

- Static HTML page with table/charts is ok. Visuals should be clean and well-organized.
- You can use Streamlit for quick Python dashboards
- Host on Vercel/Render/Replit (free tier) or other publicly accessible URL of your choice

This should be the actual live webpage, not a static GitHub repo link to the source code.

Part 4: Explainers
Please add an explainer section to your site explaining how you would productionize your pipeline given this tech stack:

- n8n (workflow automation)
- PostgreSQL (running in Docker)
- Google Sheets (where execs do financial analysis)
- Alpha Vantage API (5 calls/min, 25/day limit)

Please answer the following questions

1. How would you schedule your code to run monthly? n8n workflow? Cron job? Something else? Show a simple workflow diagram or pseudocode
2. How would you handle the API rate limit for 100 companies? Your code works for 3 companies locally. What changes for 100 companies hitting the 25 calls/day limit?
3. How would execs access this data in Google Sheets? Direct Postgres connection? Export to CSV? Sync to BigQuery first? Justify your choice with pros/cons
4. What breaks first and how do you know? What monitoring/alerts would you add? How do you detect bad data from the API?

Deliverables

1. Code repository (GitHub/GitLab)
    a. Include README.md with setup instructions
2. Public dashboard URL
    a. Include visualizations from Part 3
    b. Include answers to Part 4

Evaluation Criteria

- Code quality: Clean, readable, maintainable
- Financial literacy: Correct metric calculations, understanding of statement relationships
- Database design: Normalization, indexing strategy, scalability considerations
- Error handling: API failures, missing data, edge cases
- Automation thinking: Practical approach to production deployment

Notes

- Use Alpha Vantage free tier (5 API calls/min, 25/day)
- Use PostgreSQL however you prefer (local install, Docker, cloud free tier, etc). Include clear setup instructions in your README

Good luck and have fun! :)