10EQS Evaluation

60-Minute Data Integration Challenge

The Challenge

Build a tool that helps a small business owner track their product pricing against market conditions. Your tool should:
1. Read the provided product data
2. Integrate data from an external source of your choice
3. Generate actionable insights

Sample Data (see full products.csv below)

```
product_name,our_price,category
Coffee Beans (1lb),14.99,Beverages
Green Tea (50 bags),8.99,Beverages
Chai Latte Mix,9.99,Beverages
```

Requirements

1. Read and process the CSV data

2. Choose and integrate ONE external data source
   - Could be pricing data, market trends, economic indicators, etc.
   - Document why you chose this source

3. Create ONE useful insight
   - Examples: price comparison, trend alert, recommendation system
   - The specific insight is up to you - be creative!

4. Push your work to a Github repository (make sure it's public!) and submit your link

Submission Requirements

Repository Structure

```
├── README.md           # Setup & analysis documentation
├── data/
│   └── products.csv    # Original data file
├── src/
│   ├── analysis.py     # Your main script
│   └── utils.py        # Any helper functions
├── requirements.txt    # or package.json
└── report.md          # Your data insights
```

Required Elements

1. **Working Code**
   - Must run with a single command (document it in README)
   - Include all requirements/dependencies
   - Handle API keys via environment variables (include .env.example)
2. **README.md**
   - Clear setup instructions
   - Required API keys/credentials
   - Brief explanation of your approach
   - Any known issues or limitations
   - Time spent on each component
3. **report.md**
   - Summary of data cleaning steps
   - Key insights discovered
   - Visualization (if any)
   - Recommendations based on findings

- External data source documentation

Running the Solution

Your code should:

- Accept the CSV file path as an argument
- Output analysis to report.md
- Handle errors gracefully
- Run in under 2 minutes

Example run command:

```
# Python
python src/analysis.py data/products.csv

# JavaScript
node src/analysis.js data/products.csv
```

Report Format

The generated report.md should include:

1. Data quality issues found
2. Cleaned data summary
3. External data integration results
4. Business insights
5. Future recommendations

Keep the output clear and business-focused. Imagine explaining your findings to a small business owner.

Tips

- Use git commits to show your work progression
- Include comments for any assumptions made

- Document any data quality issues found

CSV Data

```
product_name,our_price,category,current_stock,restock_threshold,
"Organic Coffee Beans (1lb)",14.99,Beverages,45,25,2024-11-15,Be
"Premium Green Tea (50 bags)",8.99,Beverages,32,20,2024-11-10,Te
"Masala Chai Mix (12oz)",9.99,beverages,18,15,2024-11-18,Spice \
"Yerba Mate Loose Leaf (1lb)",$12.99,Beverages,5,10,2024-11-01,S
"Hot Chocolate Mix (1lb)",7.99,Beverages,50,30,11/12/2024,Sweet
"Earl Grey Tea (100 bags)",11.99,beverages,28,25,2024-11-14,Tea
"Espresso Beans (1lb)",16.99,Beverages,22,20,2024-11-16,Bean Bro
"Chamomile Tea (30 bags)",6.99,Tea,12,15,2024-11-05,Tea Time Imp
"Matcha Green Tea Powder (4oz)",19.99,Beverages,8,,2024-11-17,Te
"Decaf Coffee Beans (1lb)",15.99,Beverages,15,15,2024-11-13,Bean
"Mint Tea (25 bags)",7.49,Beverages,out of stock,12,2024-10-30,T
"Instant Coffee (8oz)",11.99,coffee,25,20,2024-11-19,"Coffee Co
"Rooibos Tea (40 bags)",,Beverages,30,20,2024-11-08,Tea Time Imp
"cold brew concentrate",13.99,Beverages,19,15,2024-11-20,Bean Br
```