FRE 521D: Data Analytics in Climate, Food and Environment

Assignment 2: ETL Pipeline and Data Enrichment

Automating Climate Data Integration for Agricultural Analysis

Course	FRE 521D - Winter 2026
Instructor	Asif Ahmed Neloy
Released	January 16, 2026, at 12:30 PM
Due Date	**February 2, 2026, at 11:59 PM PST**
Weight	10% of final grade
Prerequisites	Completed Assignment 1 database
Submission	Canvas (one submission per team)

1. Context

Your climate-agriculture database from Assignment 1 provides a solid foundation, but real-world analysis requires richer data. The research team has identified a critical gap: the current temperature data provides only annual anomalies, but agricultural production is heavily influenced by weather patterns during specific growing seasons. Monthly and seasonal weather variations, precipitation levels, and extreme temperature events all play crucial roles in determining crop outcomes.

To address this gap, you will build an automated Extract-Transform-Load (ETL) pipeline that enriches your database with detailed historical weather data from the Open-Meteo API. This API provides free access to historical weather observations including daily temperature, precipitation, and other variables at specific geographic coordinates.

The challenge extends beyond simply fetching data. You must design a pipeline that handles API rate limits gracefully, transforms raw JSON responses into your relational schema, validates incoming data against quality rules, and can be scheduled to run automatically. The pipeline you build will directly support the integrated analysis in your Final Project.

2. Learning Objectives

Upon completion of this assignment, you will be able to:

1. Design and implement ETL pipelines that extract data from REST APIs
2. Handle common API challenges including authentication, pagination, and rate limiting
3. Transform semi-structured data (JSON) into relational database tables
4. Implement data validation and error handling in automated workflows
5. Document data lineage and create reproducible data pipelines

3. Data Source: Open-Meteo Historical Weather API

3.1 API Overview

Open-Meteo is a free, open-source weather API that provides historical weather data without requiring authentication or API keys. This makes it ideal for educational purposes and prototyping.

Base URL: https://archive-api.open-meteo.com/v1/archive

Documentation: https://open-meteo.com/en/docs/historical-weather-api

Rate Limit: 10,000 requests per day (approximately 7 requests per minute to stay safe)

3.2 Required Parameters

Your API requests must include the following parameters:

- **latitude, longitude:** Geographic coordinates for the location (use country centroids provided below)
- **start_date, end_date:** Date range in YYYY-MM-DD format
- **daily:** Comma-separated list of variables to retrieve

3.3 Required Weather Variables

You must extract the following daily variables for each country:

Variable	API Parameter	Unit
Mean Temperature	temperature_2m_mean	Celsius
Maximum Temperature	temperature_2m_max	Celsius
Minimum Temperature	temperature_2m_min	Celsius
Precipitation Sum	precipitation_sum	mm
Rain Sum	rain_sum	mm
ET0 Reference Evapotranspiration	et0_fao_evapotranspiration	mm

3.4 Country Coordinates

A CSV file named **country_centroids.csv** will be provided on Canvas. This file contains the latitude and longitude for major agricultural countries that you must query. You are required to extract weather data for at least 30 countries from this list.

4. Tasks and Deliverables

Task 1: Pipeline Architecture Design (15 points)

Before writing code, design your ETL pipeline architecture. Your design should address data flow, error handling, and operational considerations.

Requirements:

a) Create a data flow diagram showing extraction, transformation, and loading stages
b) Define the schema for new tables that will store the weather data
c) Specify how the new weather data will relate to existing tables from Assignment 1
d) Document your strategy for handling API rate limits and failures
e) Describe how you will track data lineage (source, extraction time, transformation steps)

Deliverable: A PDF document (maximum 3 pages) containing your architecture diagram and written explanations.

Task 2: ETL Pipeline Implementation (30 points)

Implement a Python-based ETL pipeline that extracts weather data from the Open-Meteo API and loads it into your MySQL database.

Requirements:

f) Extract daily weather data for at least 30 countries for the years 2015-2023
g) Implement rate limiting to avoid exceeding API quotas (minimum 5 second delay between requests)
h) Handle API errors gracefully with retry logic (at least 3 retries with exponential backoff)
i) Transform JSON responses into normalized database records

j) Log all extraction activities including successes, failures, and record counts
k) Make the pipeline idempotent (running it twice should not create duplicate records)

Deliverable: A Jupyter notebook or Python script containing your complete ETL pipeline with documentation.

Task 3: Data Aggregation and Integration (15 points)

The raw daily weather data must be aggregated and integrated with your crop production data from Assignment 1.

Requirements:

l) Create aggregated views for monthly and annual weather summaries by country
m) Calculate derived metrics: growing degree days, precipitation variability, extreme temperature days
n) Create an integrated view that combines weather data with crop production data at the country-year level
o) Document any assumptions made in matching weather data to crop data

Deliverable: SQL scripts for all aggregation and view creation, plus a brief explanation of your derived metrics.

Task 4: Business Questions (30 points)

Using your enriched dataset, answer the following business questions. These questions require you to combine data from multiple sources and apply the analytical concepts from class. You can either use python or SQL to answer questions.

Question 1 (8 points): Growing Season Weather Patterns

Agricultural advisors need to understand weather patterns during critical growing seasons. For the major wheat-producing countries, analyze the relationship between growing season precipitation (April-August for Northern Hemisphere, October-February for Southern Hemisphere) and wheat yields from 2015-2023. Identify countries where precipitation variability appears to have the strongest impact on yields. Present your methodology clearly, including how you determined which hemisphere each country belongs to.

Question 2 (8 points): Extreme Weather Events and Production Shocks

Insurance companies and commodity traders want to understand how extreme weather affects production. Define "extreme" weather years (your choice of threshold, but justify it) and analyze whether extreme weather years correspond to production declines. Calculate the average production loss during extreme years compared to normal years for at least three major crops. Discuss the limitations of using country-level weather data for this analysis.

Question 3 (7 points): Water Balance and Irrigation Dependencies

Water resource planners need to identify regions where agriculture is becoming increasingly dependent on irrigation due to changing rainfall patterns. Using precipitation and evapotranspiration data, calculate a simple water balance indicator for each country-year. Compare this indicator against irrigation percentages from your crop data. Which countries show increasing water deficits alongside low irrigation infrastructure? These represent potential food security risks.

Question 4 (7 points): Data Pipeline Reliability Assessment

Before relying on this data for operational decisions, stakeholders need to understand the reliability of your data pipeline. Create a comprehensive data quality report that includes: completeness metrics (what percentage of expected country-year-variable combinations were successfully extracted), consistency checks (do aggregated values match source data), and timeliness metrics (how long did extraction take, were there failures). Recommend improvements for production deployment.

Deliverable: For each question, provide: (1) your analytical approach and any SQL/Python code, (2) results with appropriate visualizations

5. Submission Format

Submit a single ZIP file named **A2_TeamName.zip** containing:

6. **pipeline_architecture.pdf** - Design documentation (Task 1)
7. **etl_pipeline.py** or **etl_pipeline.ipynb** - Implementation (Task 2)
8. **aggregation.sql** or **aggregation.ipynb**- Aggregation and integration scripts (Task 3)
9. **business_questions.ipynb** or **business_questions.sql** - Analysis with visualizations (Task 4)

6. Evaluation Rubric

Criterion	Excellent	Satisfactory	Needs Work	Points
Architecture	Clear diagrams, comprehensive error handling plan	Functional design, some gaps	Incomplete or unclear design	15
ETL Pipeline	Robust, handles errors, idempotent, well-logged	Works but brittle, limited error handling	Does not complete, poor structure	30
Aggregation	Well-designed metrics, clean integration	Basic aggregation, minor issues	Missing metrics, poor integration	15
Business Questions	Insightful analysis, clear visualizations	Correct but surface-level analysis	Incomplete or incorrect analysis	30
Code Quality	Modular, documented, follows best practices	Readable, some documentation	Messy, no documentation	10
TOTAL				**100**

7. Academic Integrity

This is a team assignment. All code must be your own team's work. You may reference online tutorials and documentation, but must cite sources for any adapted code. AI assistants may be used for debugging and syntax help; document their use if substantial. Sharing API responses or processed data with other teams is not permitted.

8. Getting Help

If you encounter difficulties:

10. Review the Open-Meteo API documentation carefully
11. Check the course discussion board for common questions
12. Attend office hours for pipeline debugging help
13. Test your pipeline on a small subset before running the full extraction

Good luck!