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

Assignment 1: SQL Access Layer

Building a Data Foundation for Climate-Agriculture Analysis

Course	FRE 521D - Winter 2026
Instructor	Asif Ahmed Neloy
Released	January 5, 2026
Due Date	**January 14, 2026, at 11:59 PM PST**
Weight	10% of final grade
Submission	Canvas (one submission per team)

1. Context

The Food and Agriculture Organization of the United Nations has tasked your team with building a data infrastructure to support ongoing research into how climate variability affects global agricultural productivity. Your organization has received two datasets: one containing historical crop production statistics, and another containing temperature anomaly records from NOAA. Before any meaningful analysis can occur, these datasets must be properly ingested into a relational database, validated for quality, and structured to support efficient querying.

The raw data files present several challenges typical of real-world datasets: inconsistent formatting, missing values represented in multiple ways, mixed data types, and structural differences between sources. Your task is to design and implement a SQL-based data access layer that transforms these messy inputs into a reliable foundation for downstream analysis.

This assignment focuses on the critical first step of any data analytics project: ensuring that data is correctly typed, properly structured, and queryable. The work you complete here will directly feed into Assignment 2 (ETL pipelines) and the Final Project (integrated analysis).

2. Learning Objectives

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

1. Design a normalized database schema appropriate for multi-source data integration
2. Handle common data quality issues during database ingestion (type mismatches, missing values, encoding problems)
3. Write SQL queries using JOINs and CTEs to combine data from multiple tables
4. Create analysis-ready views that support business questions
5. Document schema decisions and data contracts for team collaboration

3. Provided Materials

You will receive the following files via Canvas:

3.1 crop_production_1990_2023.csv

This file contains annual crop production statistics for major cereals across countries worldwide. The data originates from FAO but has been modified to include realistic data quality issues.

Columns: Country, ISO3_Code, Region, Income_Group, Year, Crop, Area_Harvested_Ha, Production_Tonnes, Yield_Kg_Ha, Fertilizer_Use_Kg_Ha, Irrigation_Pct, Notes

Approximate rows: 25,000

Known issues: European decimal notation in some numeric columns; missing values coded as '..' or 'NA' or blank; inconsistent country naming; footnote markers embedded in numeric cells; some years stored as text

3.2 temperature_anomalies_1990_2023.csv

This file contains annual temperature anomaly data from NOAA, representing deviations from a baseline period.

Columns: Country_Name, Year, Annual_Anomaly_C, Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec

Approximate rows: 7,000

Known issues: Country names do not match crop production file exactly; some monthly values missing; negative values indicated with parentheses in some rows

4. Tasks and Deliverables

Task 1: Schema Design and Documentation (2 points)

Design a relational schema to store both datasets. Your schema should support efficient querying for climate-agriculture analysis while maintaining data integrity.

Requirements:

a) Create all tables and their relationships
b) Define appropriate primary keys and foreign keys
c) Specify data types for each column with justification
d) Document any normalization decisions (why you split or combined tables)
e) Create a country mapping table to resolve naming inconsistencies between the two source files

Deliverable: A notebook file or any other files that you have used to create them.

Task 2: Data Ingestion (3 points)

Write SQL and/or Python code to load both CSV files into your MySQL database, handling all data quality issues identified in Section 3.

Requirements:

f) All CREATE TABLE statements must include appropriate constraints (NOT NULL, CHECK, FOREIGN KEY where applicable)
g) Numeric columns must be stored as numeric types (not VARCHAR)
h) Missing values must be handled consistently (converted to NULL, not left as text placeholders)
i) Your code must be reproducible: running your scripts on a fresh database should produce identical results
j) Include comments explaining how you handled each data quality issue

Deliverable: A Jupyter notebook (.ipynb) or SQL script file containing all ingestion code with comments.

Task 3: Business Questions (5 points)

Using JOINs, CTEs, and window functions, write SQL queries to answer the following business questions. Each question requires you to think carefully about how to structure the query and interpret the results.

Question 1: Regional Production Trends

You are supporting a food supply strategy team that needs a quick, comparable snapshot of agricultural output across global regions for the most recent year in the dataset. They want to know which regions are driving total production for each crop, and whether those regions also show stronger average yields. Build a query that produces this region-and-crop summary for 2023.

Hints:

- For 2023, group by **Region** and **Crop**
- Return total production, average yield, and row count
- Sort by total production

Question 2: Climate Sensitivity by Crop

A climate risk analyst is preparing a briefing on whether crop yields tend to look better during noticeably warm years compared to typical or cooler years. They are not asking for advanced statistics yet, just a clean comparison that can be explained to non-technical stakeholders. Using the annual temperature anomaly, split observations into "warm" and "cool/normal" years and summarize average yield for each crop.

Hints:

- Join crop + temperature by country-year
- Bucket anomaly into:
    - Warm: anomaly > 0.5
    - Cool/Normal: anomaly <= 0.5
- For each crop and bucket, compute average yield and count

Question 3: Yield Gap Analysis

A development organization wants examples of strong agricultural performance in lower-income contexts to guide case studies and follow-up research. They want to identify the top yield observations in 2023 among Low income and Lower middle-income countries and see the crop and production context. Produce a ranked list of the best-performing country-crop observations.

Hints:

- Filter to 2023
- Filter to income group in (Low income, Lower middle income)
- Return country, income group, crop, yield, production
- Sort by yield desc, return top 10

Question 4: Data Quality Assessment

Before building dashboards, your team needs a quick "data risk" screen to identify countries where key numeric measures may be unreliable because they are frequently missing. Focus on the core measures used in most analysis: yield and production. For each country, compute

missing counts and missing percentages across all available records, and sort so the highest-risk countries appear first.

Hints:

- For each country:
    - total rows
    - missing yield rows and %
    - missing production rows and %
- Only include countries with at least 50 rows
- Sort by missingness

Question 5: Integrated View

Your analytics team is tired of re-writing the same join logic every time they run an analysis. They want a single analysis-ready view that combines country attributes, crop production metrics, and temperature anomaly metrics in one place. They also want a couple of simple derived fields that will be useful in future labs and the final project: a production-per-area metric and a categorical temperature bucket. Create a SQL view that standardizes this joined dataset.

Hints:

- Create view climate_agriculture_analysis
- Include:
    - country fields (name, iso3, region, income group)
    - crop fields (year, crop, production, yield, area harvested, fertilizer, irrigation, notes)
    - climate field (annual anomaly)
    - derived:
        - tonnes_per_ha
        - temp_bucket via CASE
- Use LEFT JOIN for temperature so crop rows remain even if climate is missing

Deliverable: For each question, provide your SQL query.

5. Submission Format

Submit a single ZIP file named **A1_TeamName.zip** containing all necessary files

6. Evaluation Rubric

Criterion	Excellent	Satisfactory	Needs Work	Points
Schema Design	Normalized, well-justified	Functional schema, minor issues	Missing relationships or poor normalization	15
Data Ingestion	All issues handled, reproducible, well-documented	Most issues handled, minor gaps	Data quality issues remain, not reproducible	25
Validation	Comprehensive checks, clear reporting	Basic checks completed	Incomplete or superficial validation	15
Business Questions	Correct queries, insightful interpretation	Queries work, basic interpretation	Queries incomplete or incorrect	35
Code Quality	Clean, well-commented, follows standards	Readable, some comments	Messy, hard to follow	10
TOTAL				**100**

7. Academic Integrity

This is a team assignment. You may discuss approaches with other teams at a conceptual level, but all submitted code and documentation must be your own team's work. You may use online resources and documentation, but you must cite any code snippets adapted from external sources. Use of AI assistants (such as ChatGPT or Copilot) is permitted for debugging and syntax help, but the core logic and design decisions must be your own. If you use AI tools, include a brief statement describing how they were used.

8. Getting Help

If you encounter difficulties:

6. Check the course discussion board on Canvas for common questions
7. Attend office hours (schedule posted on Canvas)
8. Post technical questions to the discussion board (do not share solution code)
9. For private concerns, email the instructor directly

Good luck!