Data Engineering Take-Home Project

Overview

At Reveel, data engineers spend much of their time working with messy, inconsistent datasets that arrive in different formats and schemas. Your task is to design and implement a small but realistic pipeline that ingests, normalizes, and analyzes a set of sample data.

This exercise is meant to evaluate how you think, how you code, and how you explain your choices. It should take around 4 hours for a strong candidate.

We are less concerned with the exact tools you choose, and more with your reasoning, clarity, and ability to wrangle messy data into useful structures.

Provided Data

You will be given datasets (attached as both CSV and PDF system exports):

- Clients — basic customer information (3 schema variants).
- Invoices — billing records tied to clients (3 schema variants).
- Rate Sheet — the standard per-shipment-type costs used for billing.

Each schema variant simulates exports from different systems, with slight differences in structure. All datasets contain realistic messiness:
- Mixed and inconsistent date formats.
- Duplicate rows and IDs.
- Missing values.
- Invalid or inconsistent foreign keys.
- Different naming conventions.

All invoice amounts are denominated in USD.

Your job is to ingest, normalize, and reconcile these datasets into a consistent model you can query.

Rate Sheet

You are provided with the following shipment type costs:

Shipment Type	Cost per Unit
GROUND	$1

2 DAY	$5
EXPRESS	$10
FREIGHT	$20

Requirements

1. Ingestion
- Load the CSV and PDF files into your pipeline.
- Your pipeline should be idempotent (running twice doesn't duplicate rows).
- Document how you chose to parse PDFs and reconcile fields across variants.

2. Transformation
- Normalize client and invoice data into consistent tables.
- Handle duplicates, invalid IDs, and inconsistent date formats gracefully.
- Decide how to treat missing or inconsistent values, and explain your choices.

3. Modeling
- Produce a fact table or equivalent dataset that ties clients and invoices together.
- Include for each invoice:
  - Client ID & name
  - Invoice ID & date
  - Invoice amount (USD)
  - Shipment type

4. Analysis Queries
Answer the following questions using your modeled data:

1. Basic: Which top 5 clients have the largest total invoice amounts outstanding?
2. Intermediate: Show the month-over-month invoice growth per client for 2024–2025.
3. Discount Scenario: Show total costs for each client, if discounts were applied:
   - 20% off GROUND
   - 30% off FREIGHT
   - 50% off 2 DAY
   - Then, who are the new top 5 spenders?
4. Reclassification Scenario: Suppose all "EXPRESS" shipments were instead billed as "GROUND" (lower cost).
   - What is the total cost savings opportunity per client?
   - Which clients have >50% savings?
   - Which clients have >$500k savings?

Deliverables

- Your code in a repo (Python, SQL, or another language of your choice).
- A short README including:
  - How to run your pipeline.
  - Assumptions you made.
  - Example outputs for the analysis queries.
  - What you would do differently if this were production code.

Notes

- Tools: You may use any libraries or tools you prefer (Pandas, SQL, dbt, PySpark, etc.).
- AI usage: You may use AI tools, but you must provide us with the prompt and be able to explain your code and reasoning in detail.
- Evaluation criteria:
  - Correctness: Does it run and produce the right results?
  - Code quality: Is it clean, modular, and easy to follow?
  - Reasoning: Are your choices explained clearly?
  - Curiosity: Did you explore edge cases and think critically about messy data?

This project is not about getting everything perfect. It's about showing us how you think, how you code, and how you deal with ambiguity.