Project Phase: Data Engineering & Quality Control

In this phase, I focused on optimizing the database structure and implementing rigorous data quality checks.

Key Points:

1. **Import Optimization:**
    - *During the data ingestion phase, I eliminated redundant columns to reduce the database footprint and improve query performance.*
2. **Text Standardization:**
    - *In the customers table, I normalized city names to uppercase using UPPER to ensure accurate geographical data aggregation.*
3. **Data Type Correction:**
    - *I altered the customer_zip_code_prefix column to VARCHAR(20) to preserve leading zeros and eliminate unwanted numerical formatting like thousand separators.*
4. **Schema Safety:**
    - *I manually adjusted VARCHAR lengths to ensure data integrity while optimizing storage efficiency.*
5. **SQL Views Documentation**:
    - *I created a view layer to separate raw data from the analytical layer. These views automatically clean names, format currencies, and prepare data for visualization.*
        - v_payments_pretty
          Cleans payment method names by replacing underscores with spaces and enforces a two-decimal currency format.
        - v_category_translation
          Maps original category names to readable English translations, removing technical underscores for reporting.
        - v_products_pretty
          Prepares a product list with sanitized category names, enabling efficient grouping and catalog analysis.
6. **Advanced Duplicate Resolution:**

   In the order_reviews table, I identified non-identical duplicates where rows shared the same review_id but differed in timestamps. I implemented a robust cleaning strategy using **Common Table Expressions (CTE)** and the

ROW_NUMBER() window function (PARTITION BY review_id ORDER BY review_creation_date DESC). This ensured that only the most recent and unique record for each review was preserved, maintaining the integrity of customer satisfaction metrics.

Future Process Optimization (Power Query):

For this specific project, I deliberately performed all transformations using SQL to maintain full control over the database logic and ensure high performance on the server side. However, for future projects involving large-scale data integration, I plan to implement **Power Query (M language)**. This will allow for more automated, "point-and-click" bulk de-duplication and faster data preparation during the final visualization phase.

7. **Comprehensive NULL Audit & Data Pruning:**

I performed a system-wide audit to identify missing values across all tables. This step was crucial for establishing a "source of truth."

Products: Detected 610 records with missing category names and metadata (including 2 records with zero physical dimensions). These were systematically excluded to prevent "Unknown" categories from skewing business insights.

Future Optimization Note: While I performed this audit and filtering manually in SQL to master the underlying data logic, this stage is significantly faster in Power Query using the "Column Quality" and "Remove Empty" features. For future large-scale projects, I will leverage Power Query's visual profiling to accelerate the data-cleaning phase.