**Detailed Analysis, Design, and Implementation Document for E-Commerce Sales Data**

**Processing with Databricks**

1. **Introduction**

This document outlines the design and implementation of data processing system using Databricks to process and analyze e-commerce sales data. The system will process raw sales data, including orders, customers, products, and transactions, and transform it into structured, analyzed datasets for business insights. We will use PySpark for all data processing tasks, focusing on performance, scalability, and data quality.

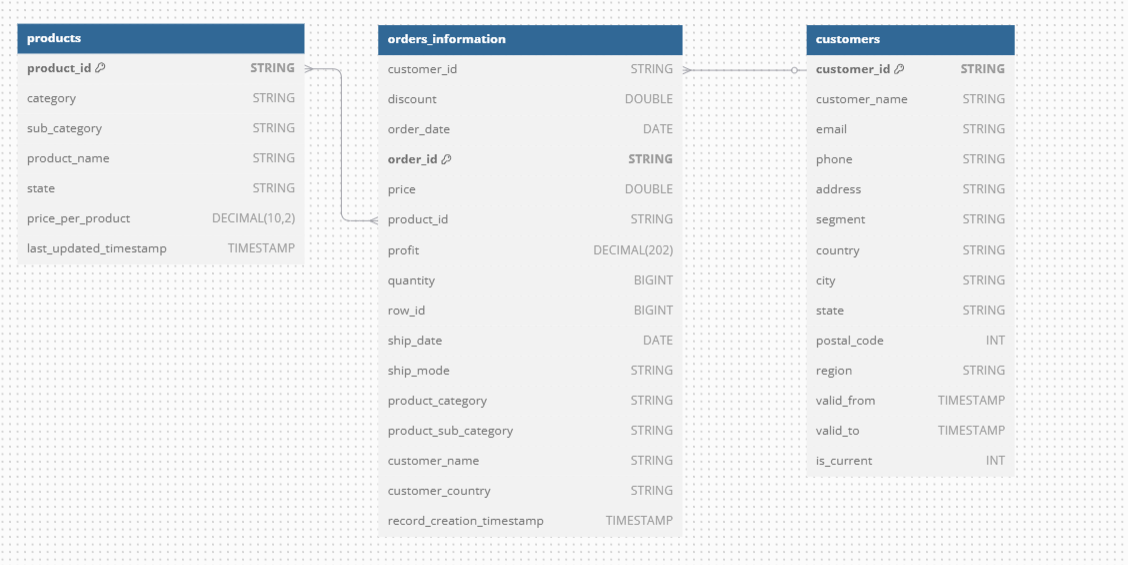
1. **Source Datasets**

Order.json: Contains details about each order, including order ID, customer ID, product ID, order date, and profit information.

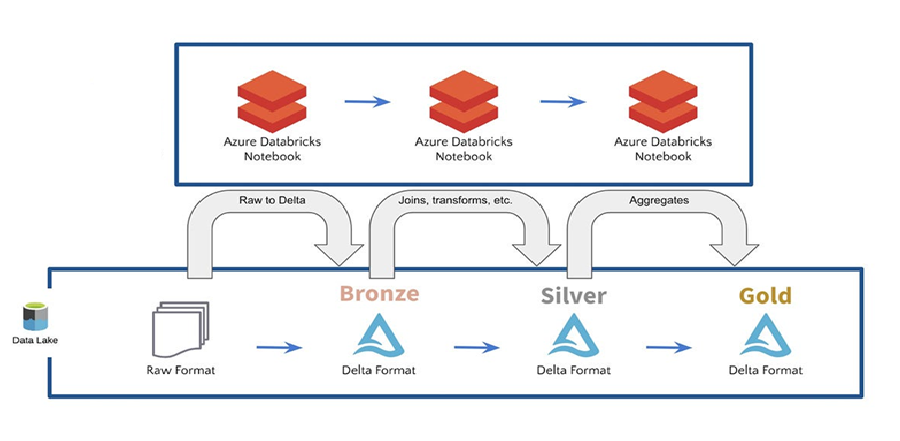
Customer.xlsx: Contains customer details such as customer ID, name, email, phone number, address, and country.

Product.csv: Contains product details including product ID, category, subcategory, product name, price, and other relevant attributes.

1. **ER Diagram**



1. **ETL Pipeline**



1. **Tasks**

*3.1 Bronze Tables Creation*

We will create bronze tables for each source dataset to store the raw data as-is. These tables will act as the foundation for any subsequent processing.

order Table: Load the Order.json file into a Spark DataFrame and create a Delta table bronze.orders.

customer Table: Load the Customer.xlsx file into a Spark DataFrame and create a Delta table bronze.customers.

product Table: Load the Product.csv file into a Spark DataFrame and create a Delta table bronze.products.

* 1. *Silver tables creation with enriched data*

Silver tables will be loaded from bronze tables and will have cleansed and enriched data and acts as an input to gold layer.

silver.products

1. Removed duplicates from products and loaded into silver.products
2. Implemented logic to handle SCD Type-1

silver.customers

1. Removed new line from email columns
2. Removed multiple new line characters from address columns
3. Removed all special characters from name column excluding single quotes between the name.
4. Filtered NaN and null as well
5. Phone number has many formats and standardized to xxx-xxx-xxxxXxxx format
6. Implemented logic to maintain SCD Type-2

silver.order\_information

1. This table will have data from customers and products as well in single place and acts as fact.
2. Cleaned date formats for order date and ship date columns and converted all date formats into yyyy-MM-dd.
   1. *Gold tables for aggregation*

gold.profit\_agg

1. This table will have aggregated data and we can use it for below dimension

* Year
* Product Category
* Product Sub Category
* Customer

1. **Exception handling**

This will help us to log any errors or issues encountered during the data loading or transformation process for debugging.

**Note:**

1. For all the data quality issues, based my assumptions have dropped duplicate data or added few default values like unknown for customer when null and unknown for products when null, but based on the actual requirements we can handle it in actual data processing.
2. Have handled minimum test cases for now and we can enhance it further based on the needs.