## **Customer 360 Data Integration**

### **Overview**

A retail business wants to build a unified Customer 360 view by integrating data from multiple sources, including online transactions, in-store purchases, customer service interactions, and loyalty programs. This project uses a mix of fact and dimension tables to ensure a clean, scalable structure.

### **Architecture Diagram:** **Tools and Technologies**

* Azure Synapse Analytics
* Azure Data Lake Storage (ADLS)
* Azure SQL Database
* Power BI

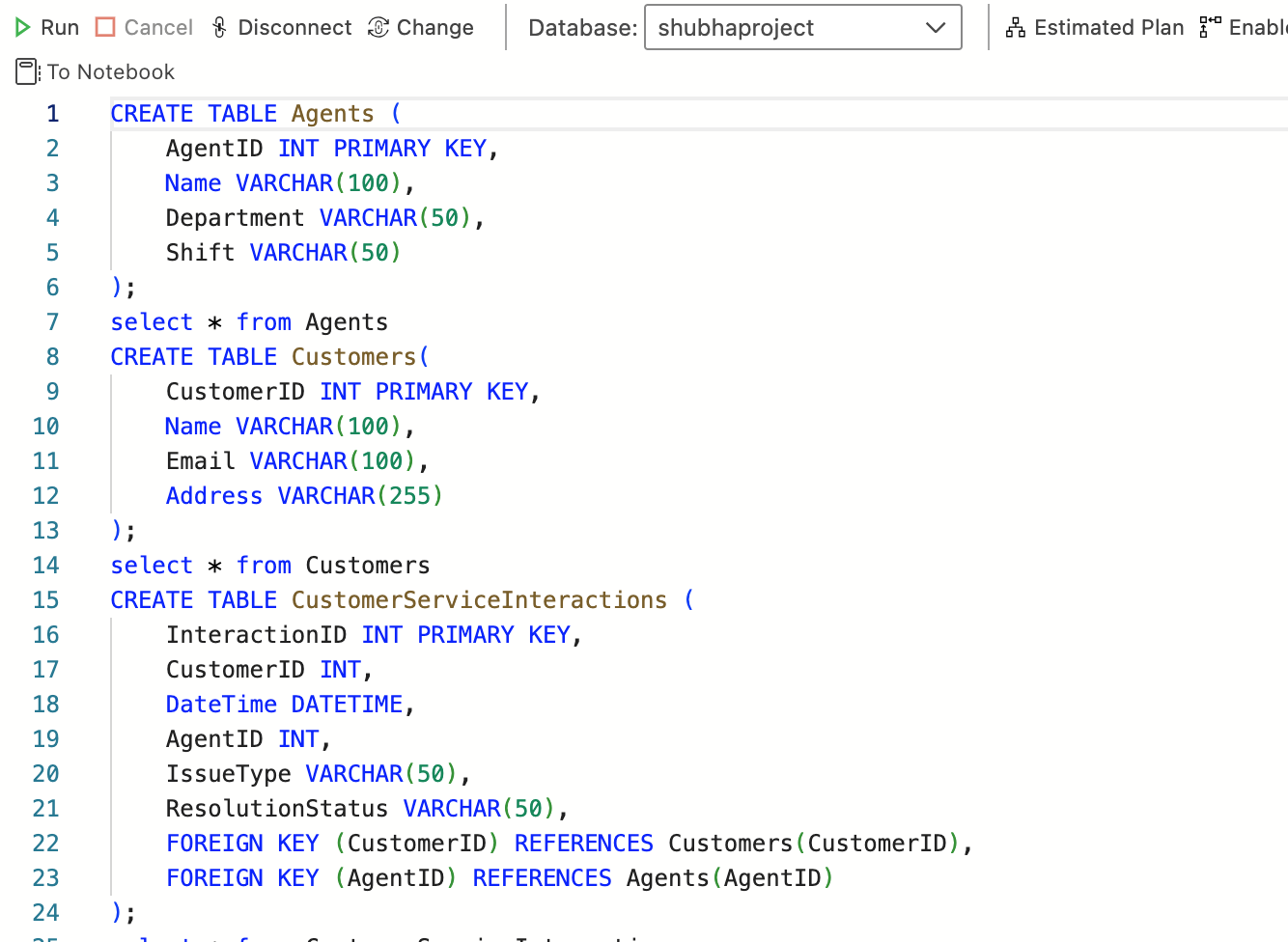
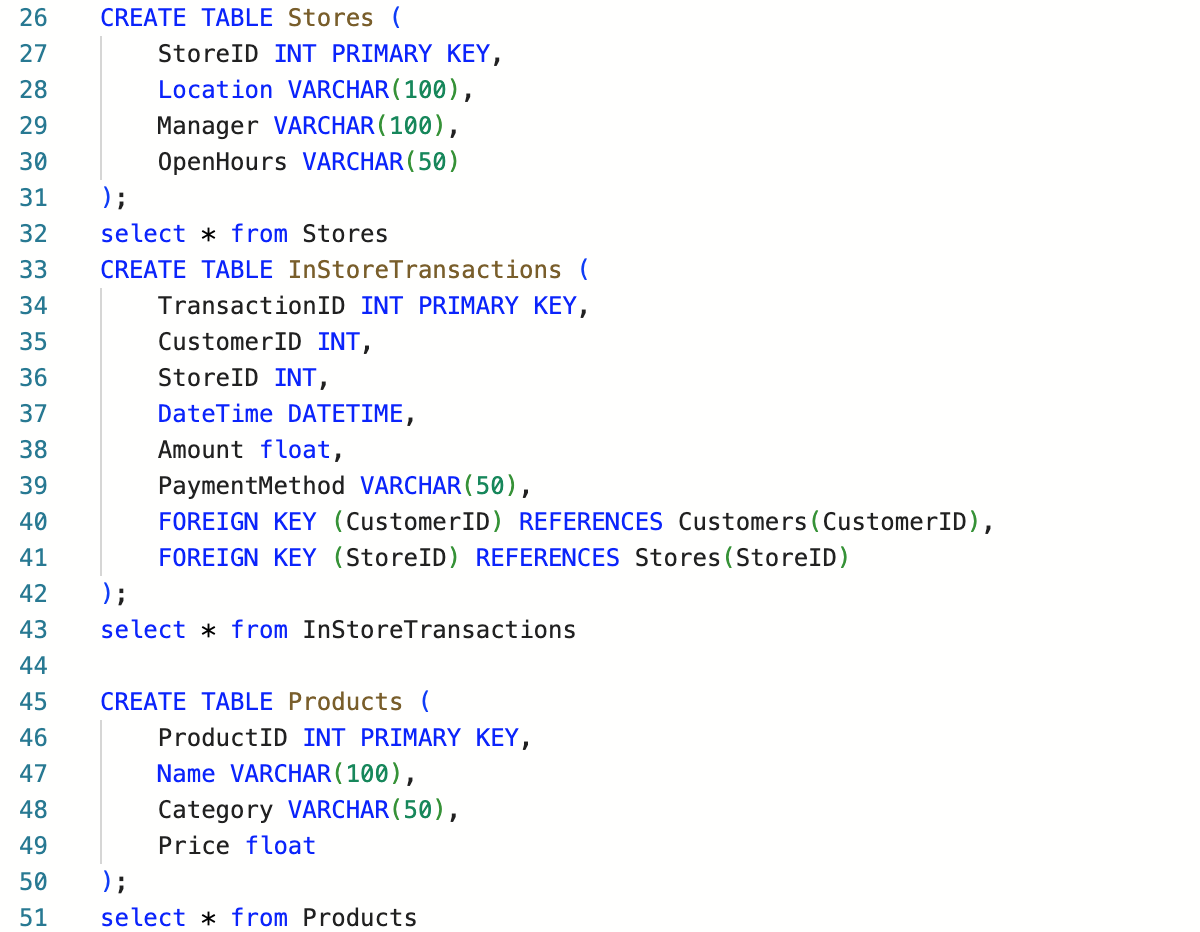
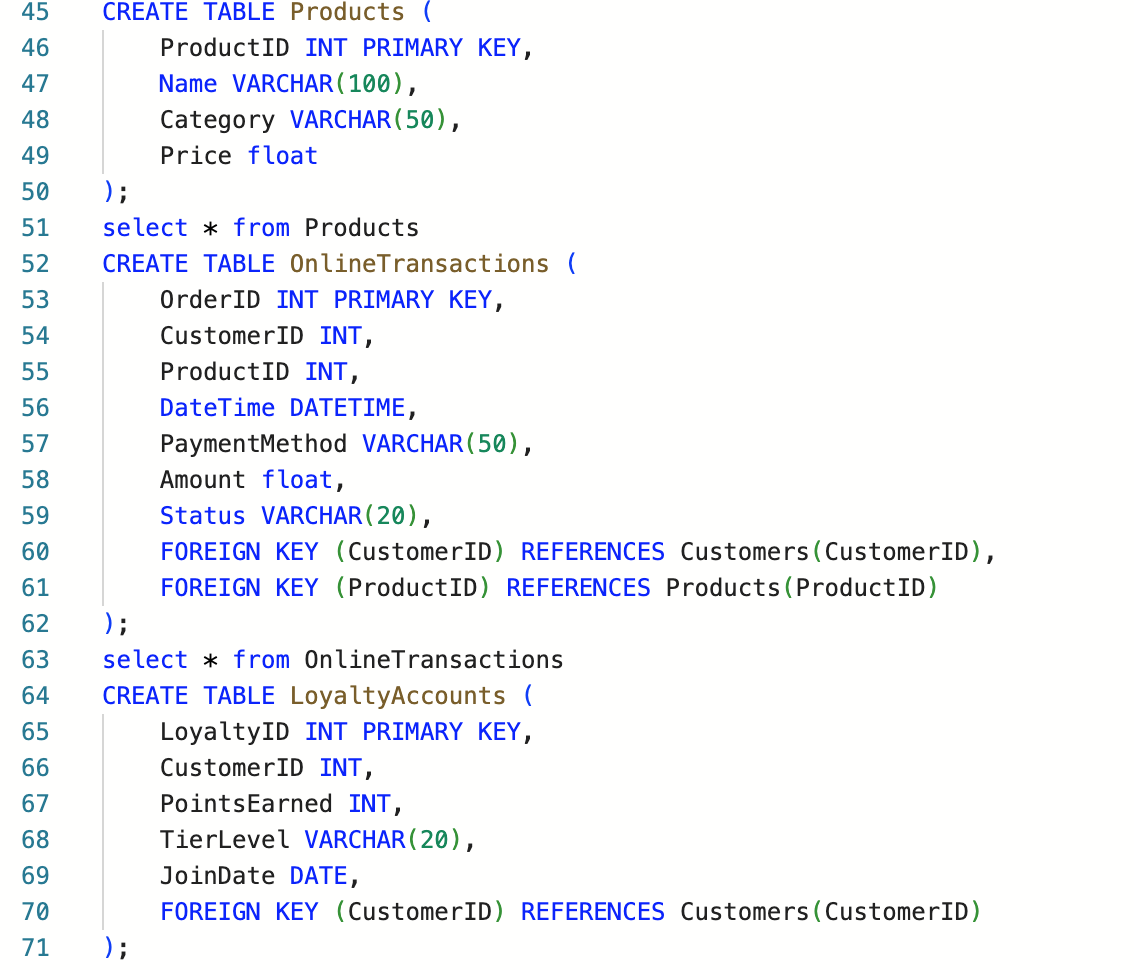
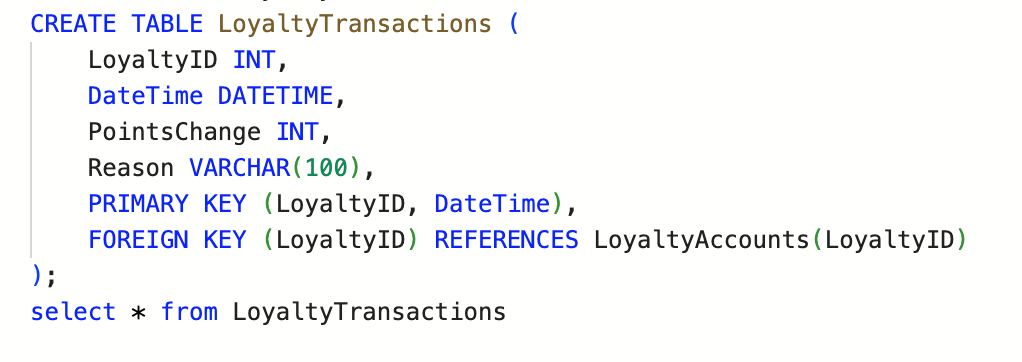
### **Step-by-Step Process**

#### **Step 1: Ingest Data**

* **Objective**: Ingest data from local system into ADLS for raw storage

### **Step 2: Create Tables in Azure SQL Database**

Once the data is ingested into ADLS Gen2, we need to create tables in **Azure SQL Database** to store the cleansed and transformed data.

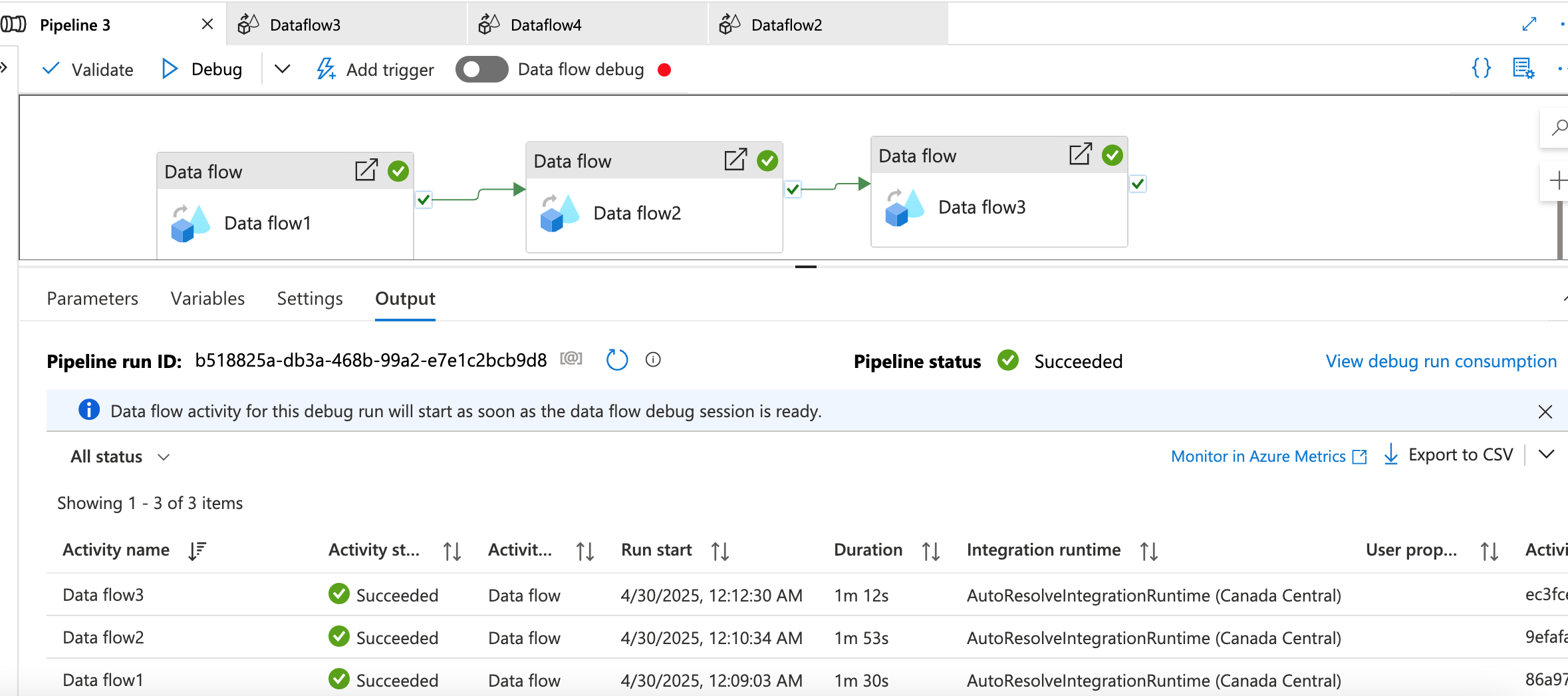
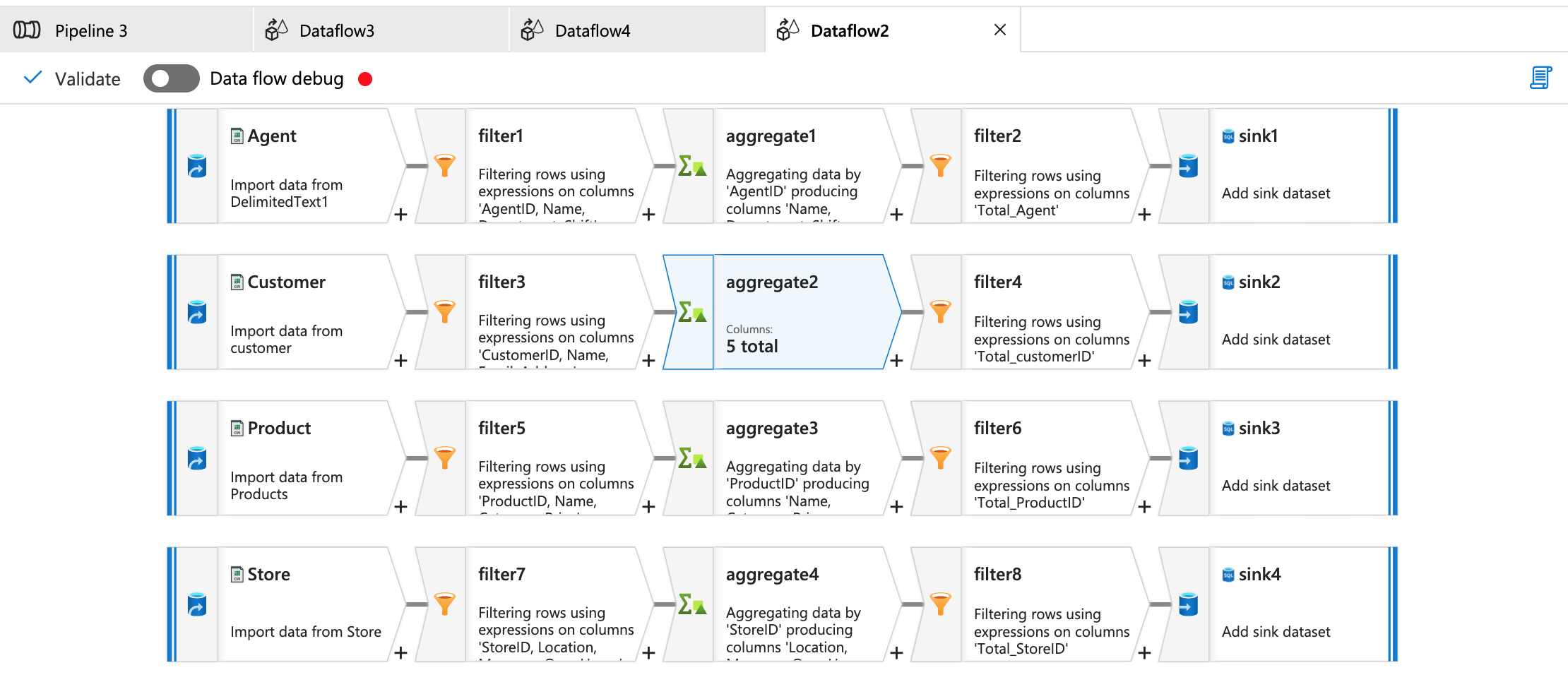
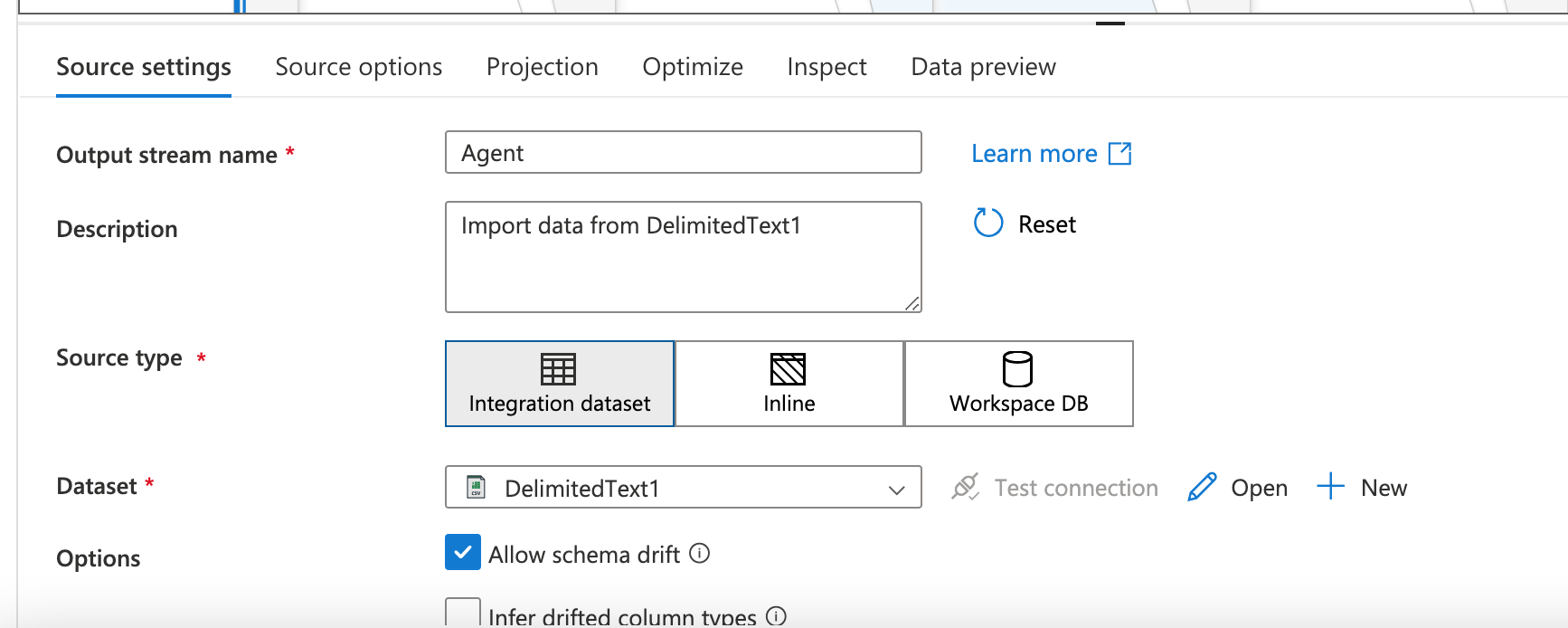
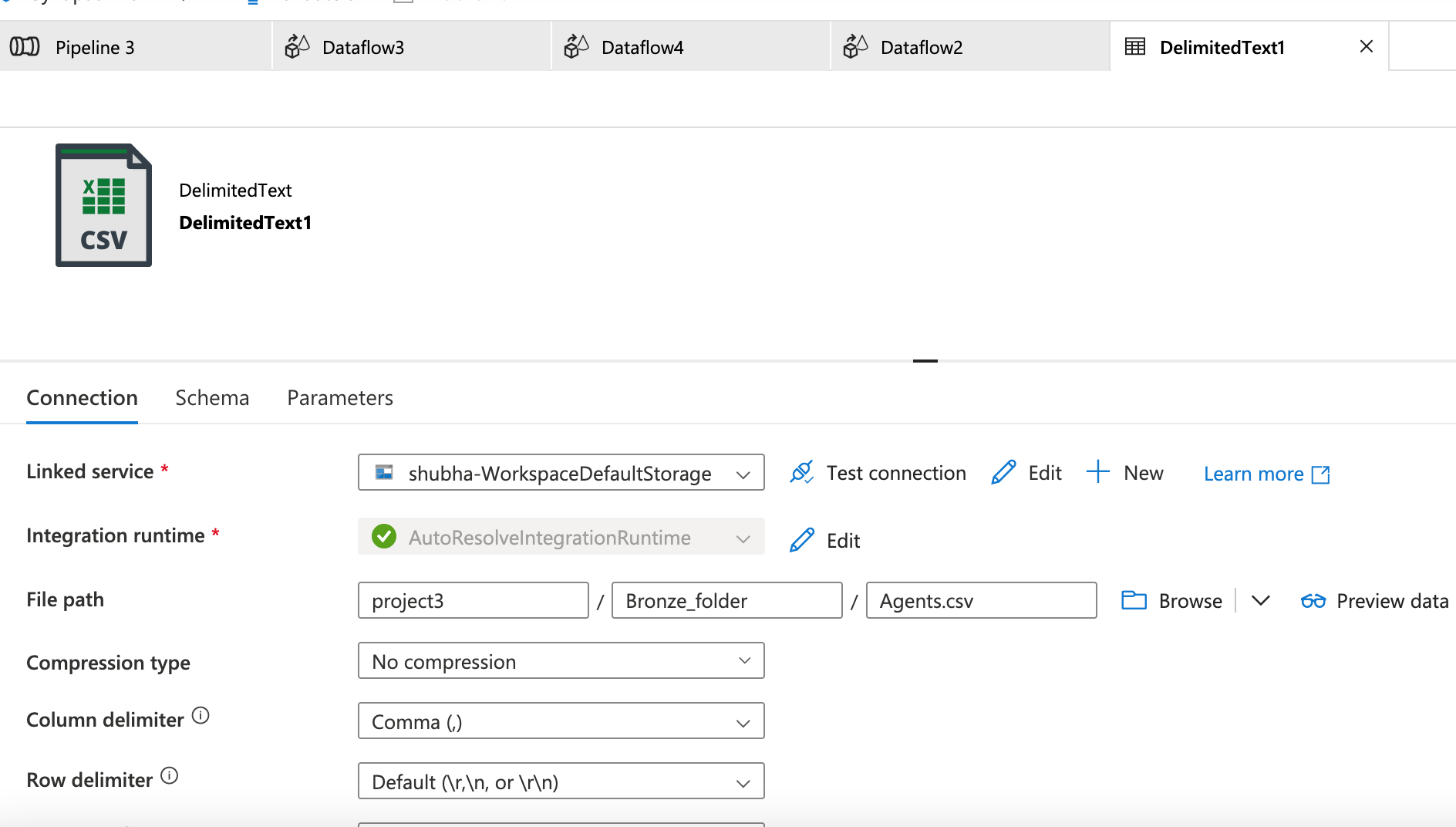
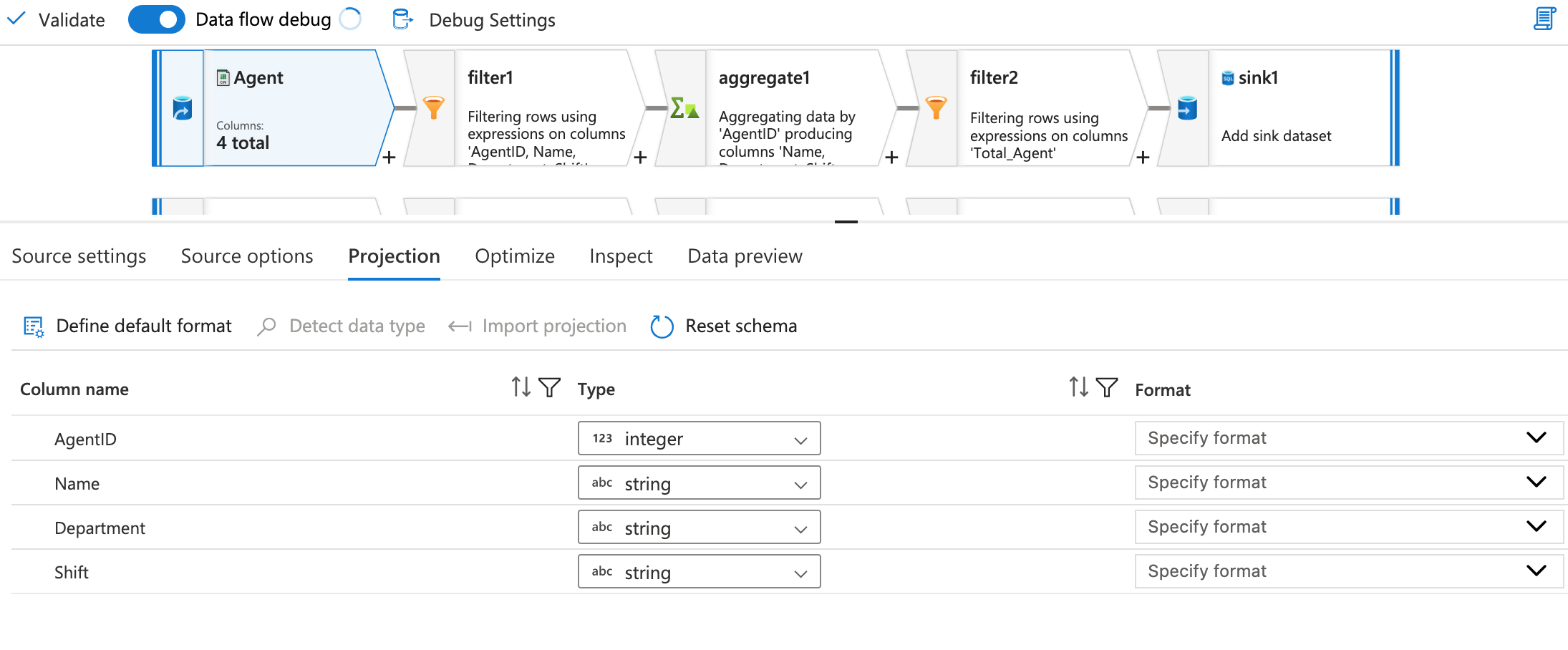
   

### **Step 3**: Clean and transform the data using **Mapping Data Flows** in ADF, then load the cleansed data into Azure SQL Database.

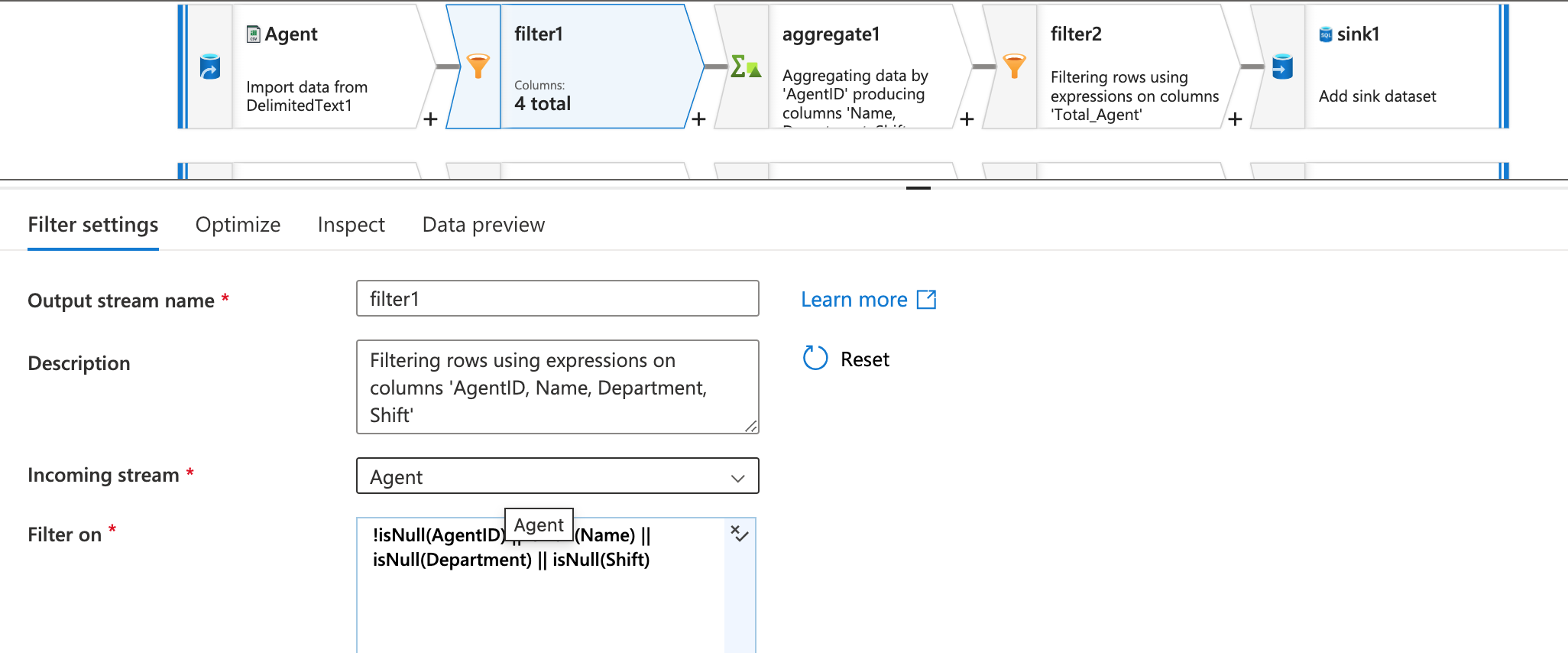
In this project, we're using three separate dataflows to handle different stages of data processing.

1. **Dataflow 1** processes independent tables (Agent, Customer, Store, Product).
2. **Dataflow 2** handles transactional data (Online Transactions, Instore Transactions, Loyalty Activity) while managing foreign key relationships.
3. **Dataflow 3** focuses on processing **Loyalty Transactions**.

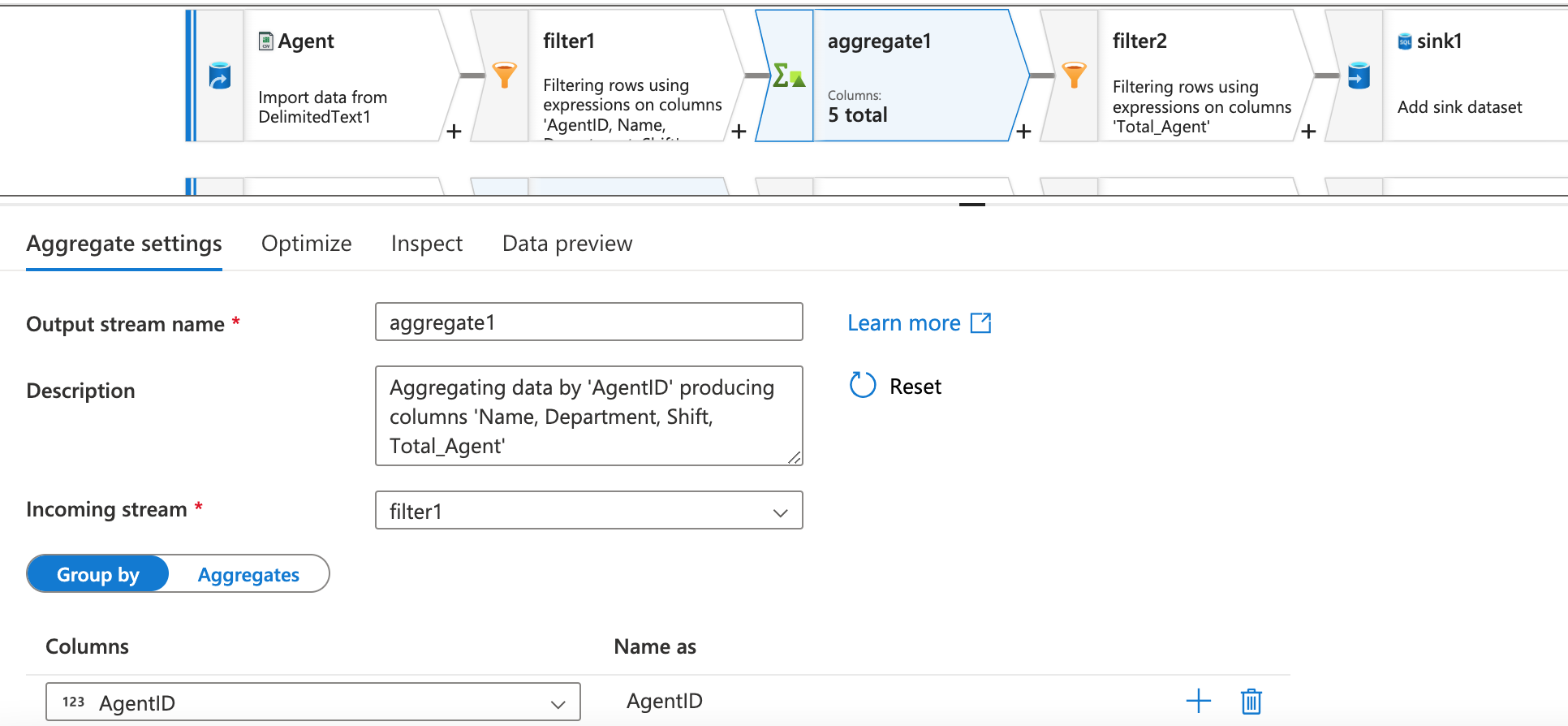
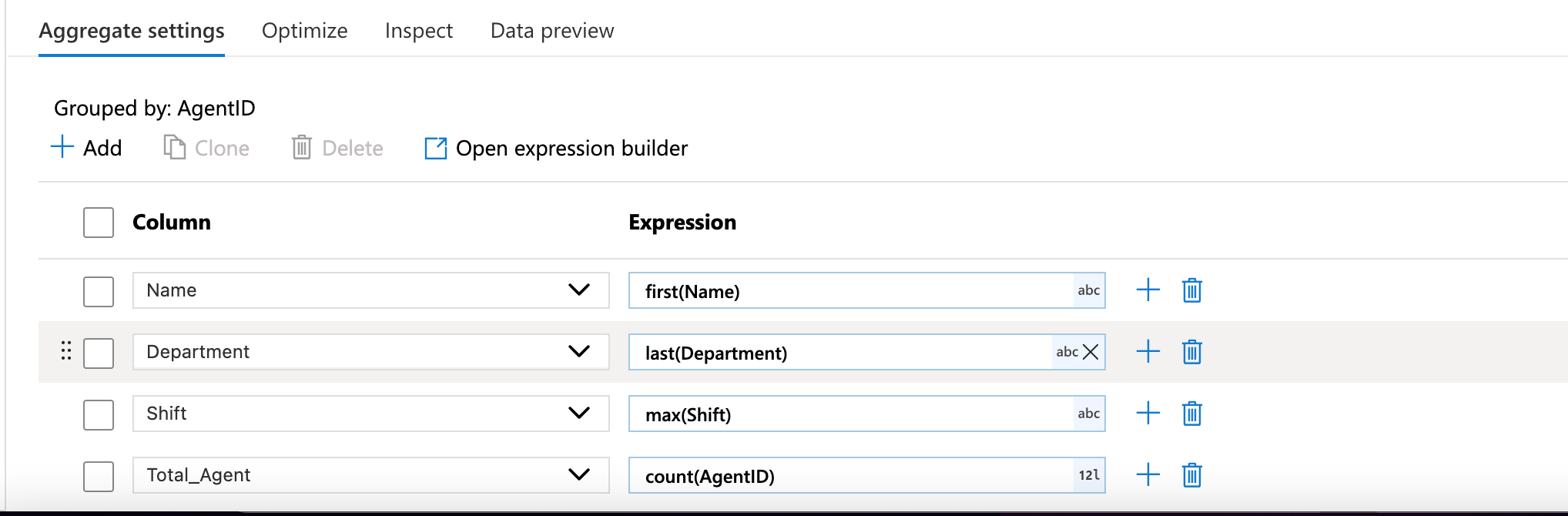
These dataflows are combined and executed in sequence to ensure a clean, structured flow of data through the pipeline.

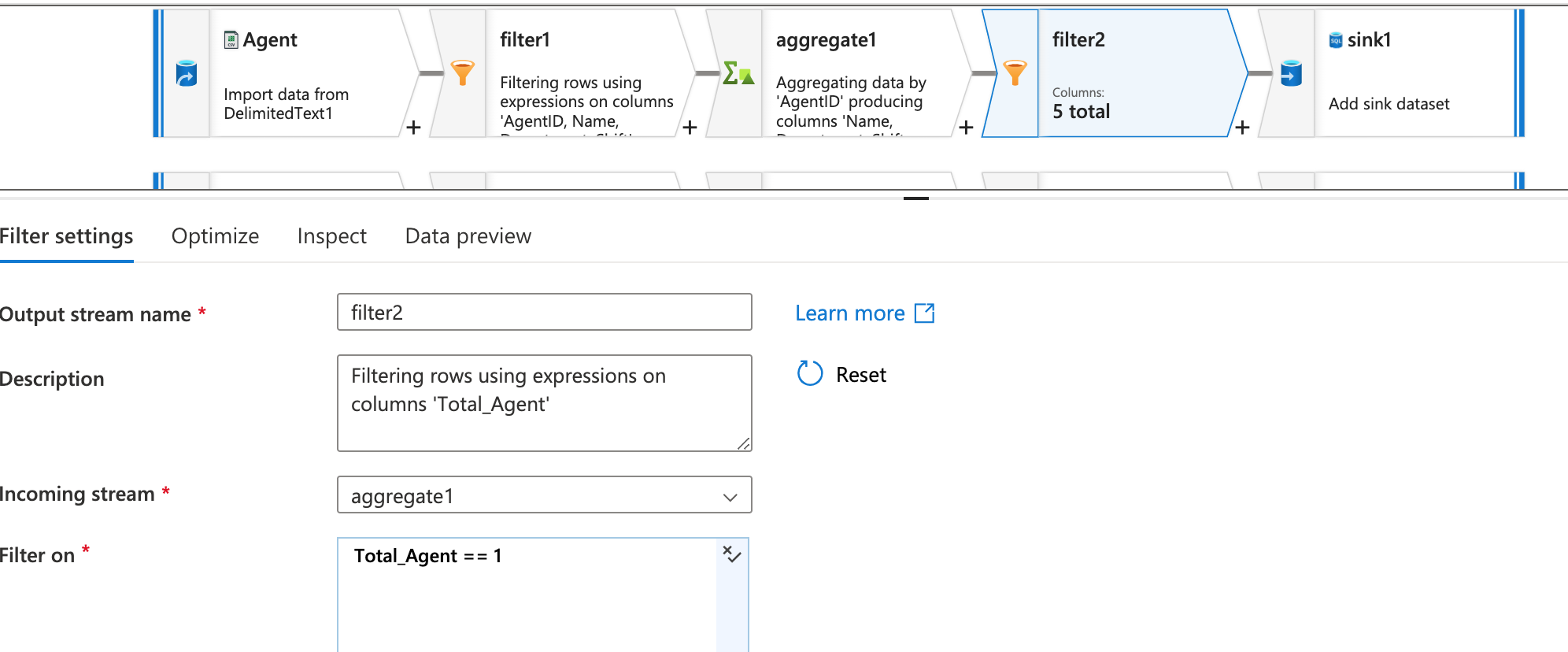
let’s go through one of the cleaning transforming flow Source:    **Filter Transformation**:

* **Purpose**: This filter removes any rows where the **AgentID**, **Name**, **Department**, or **Shift** are null.

 **Aggregate Transformation**:

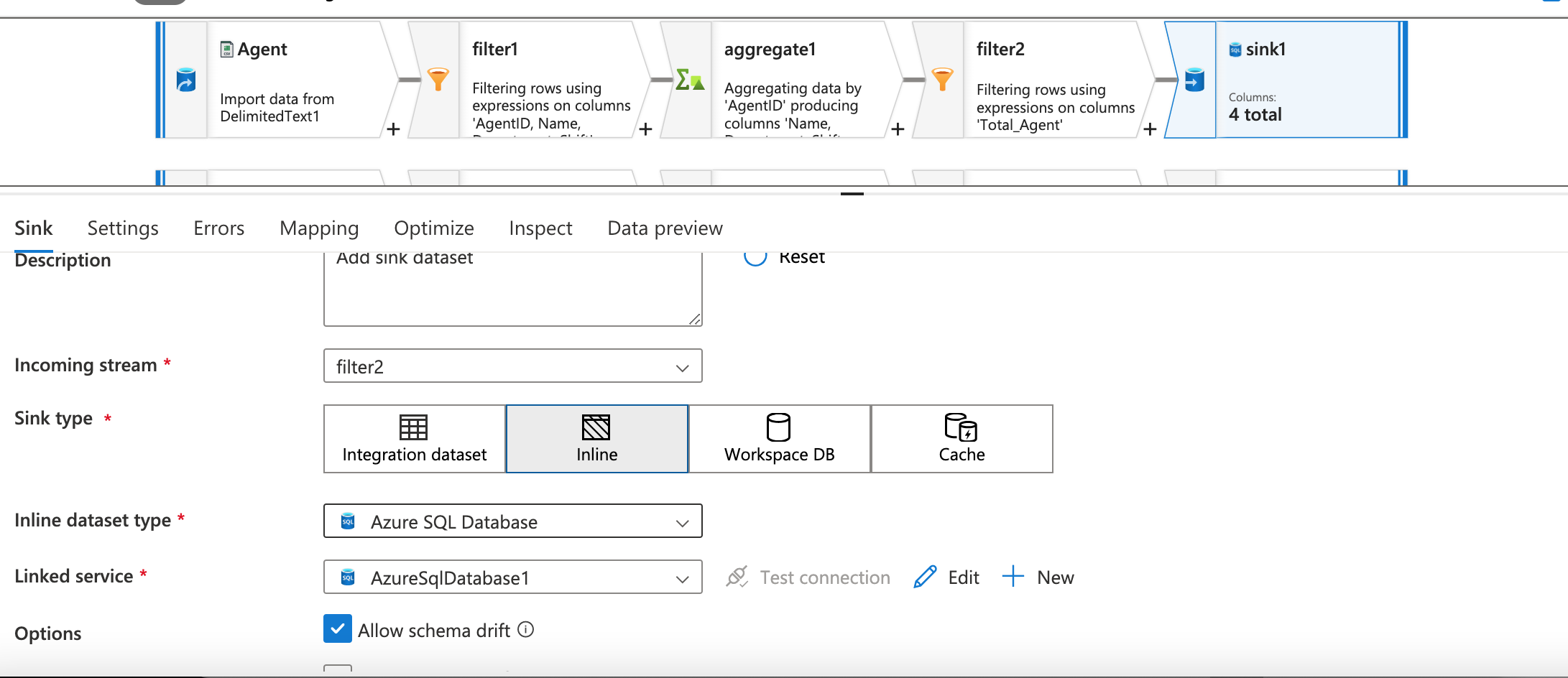
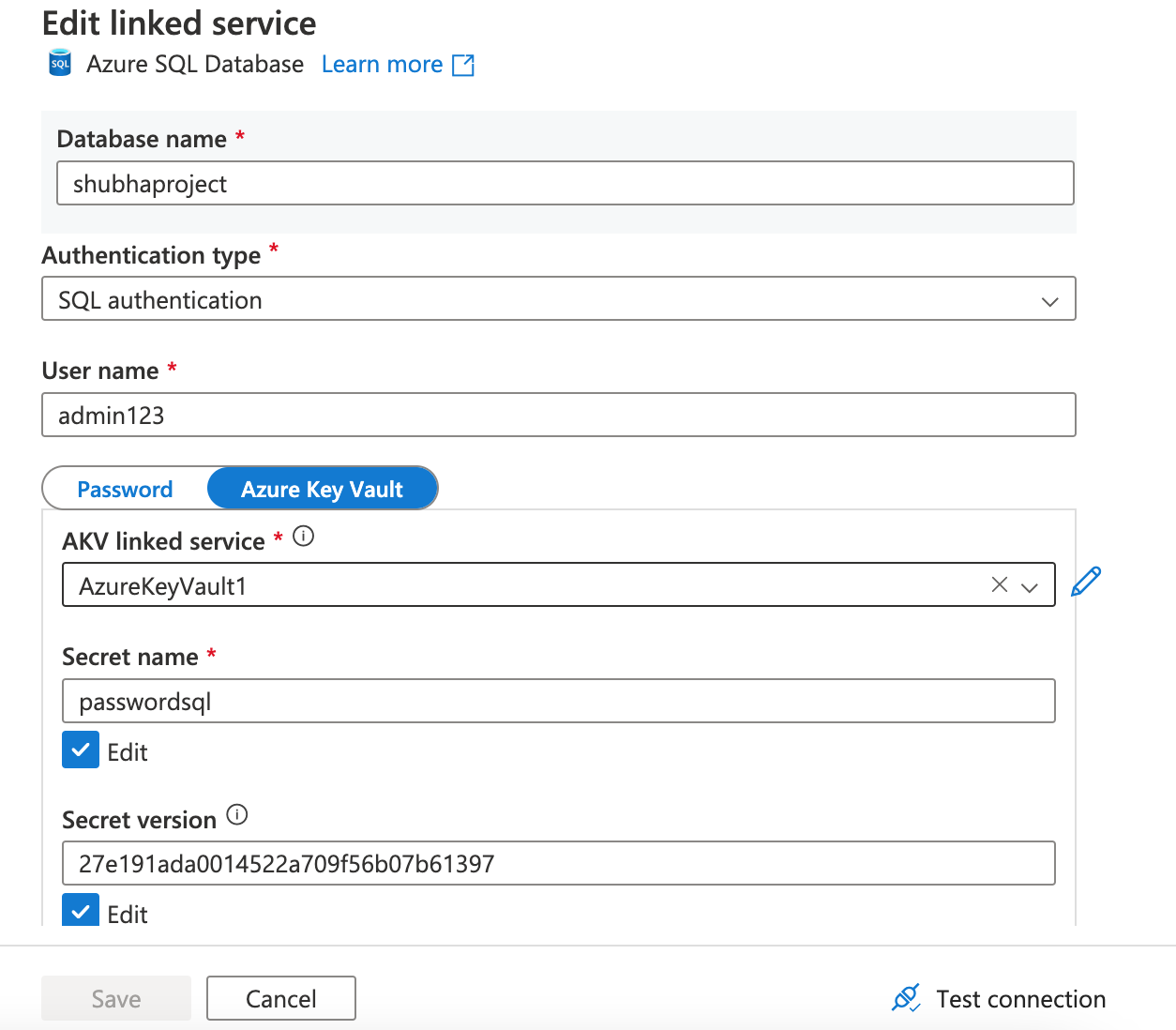
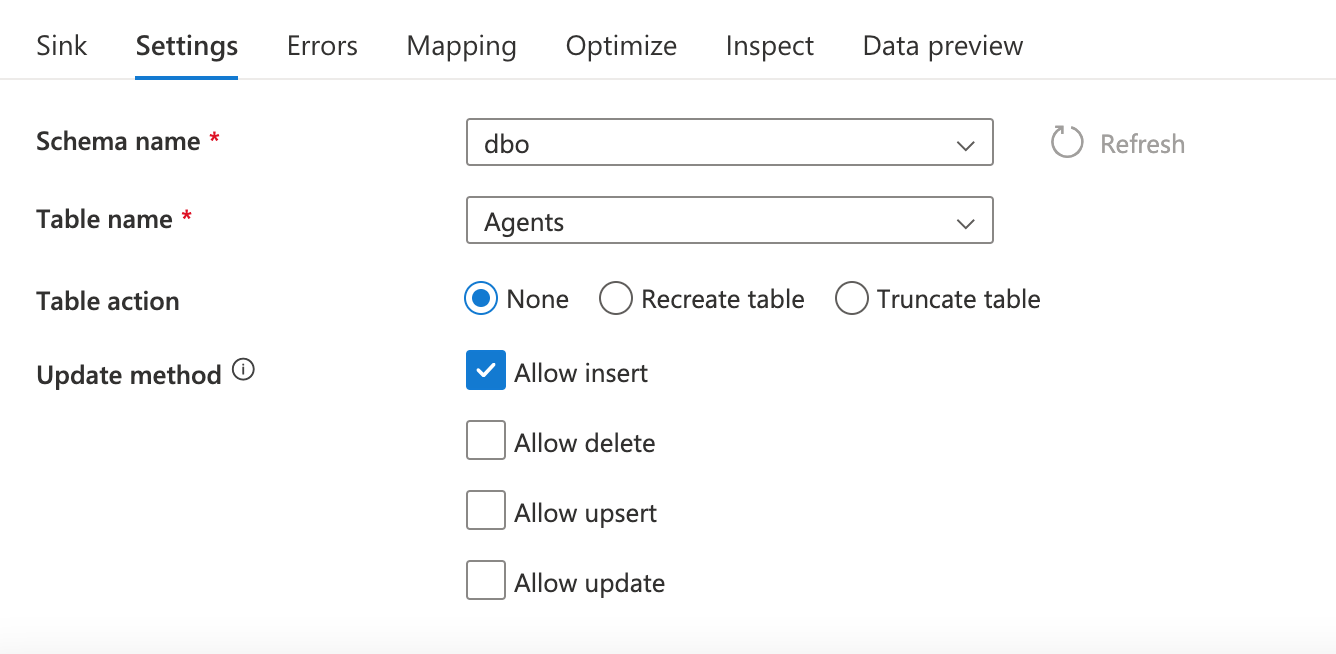
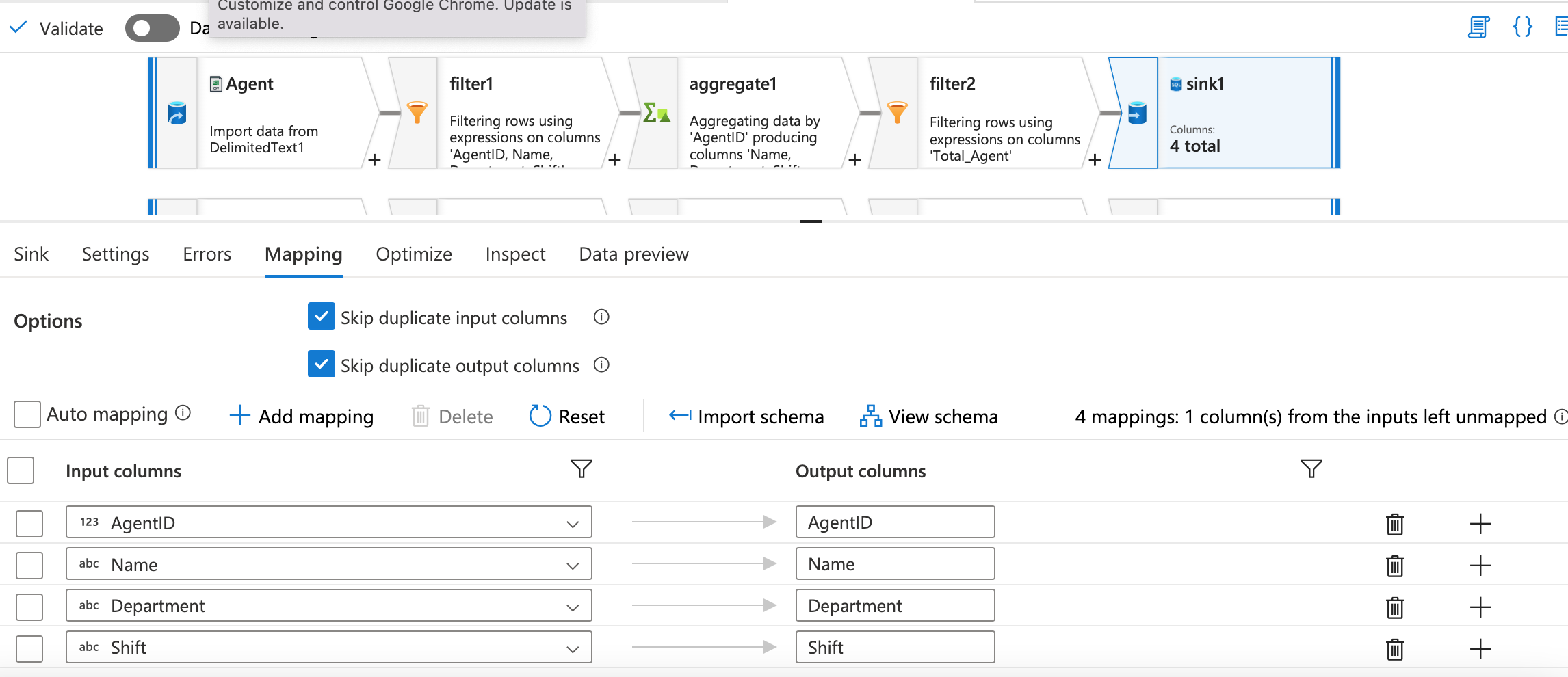
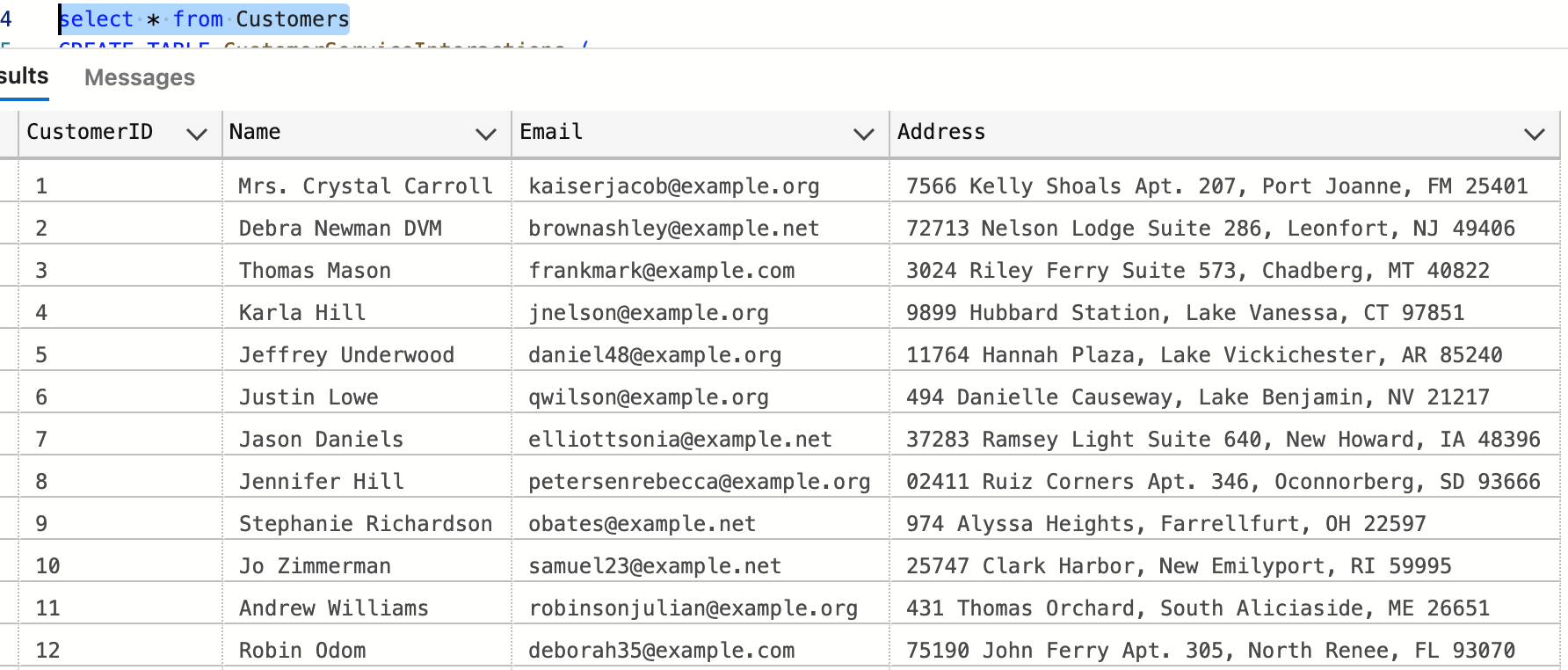
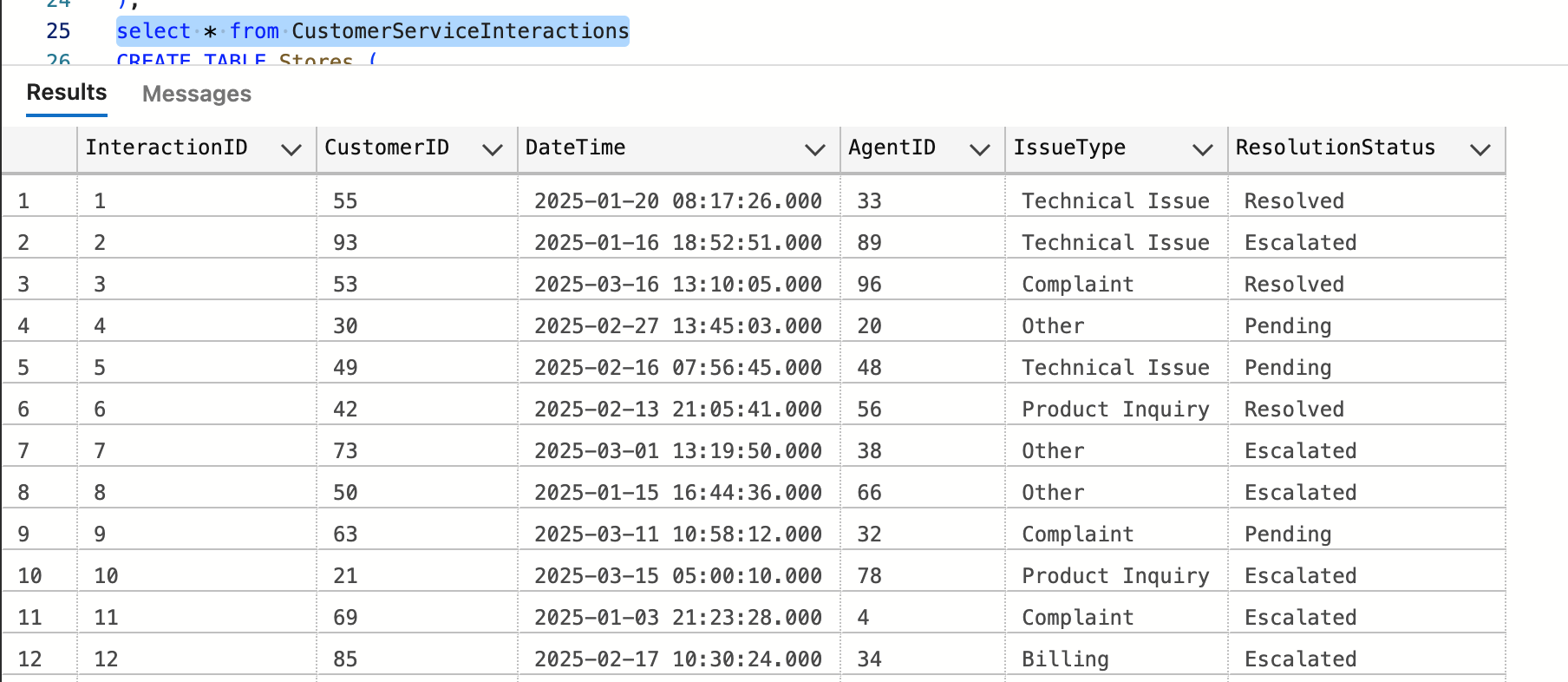
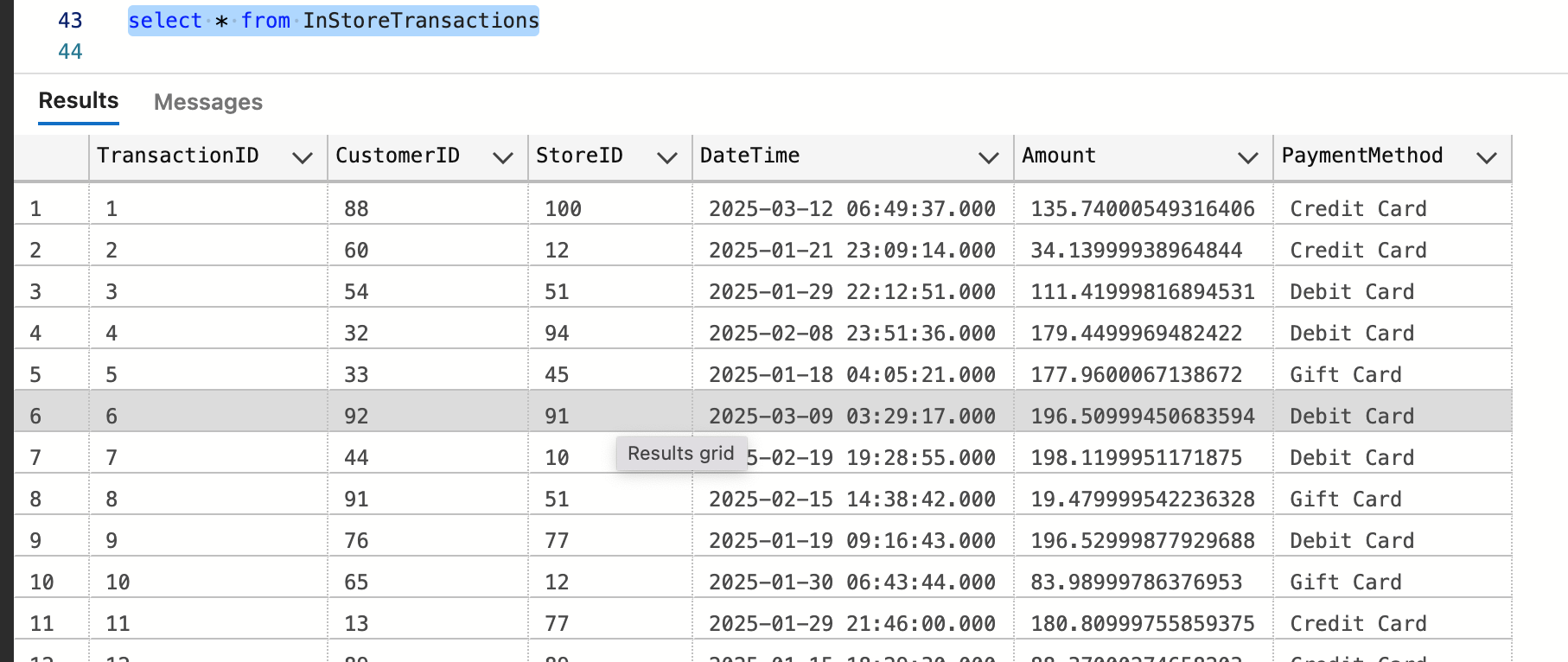
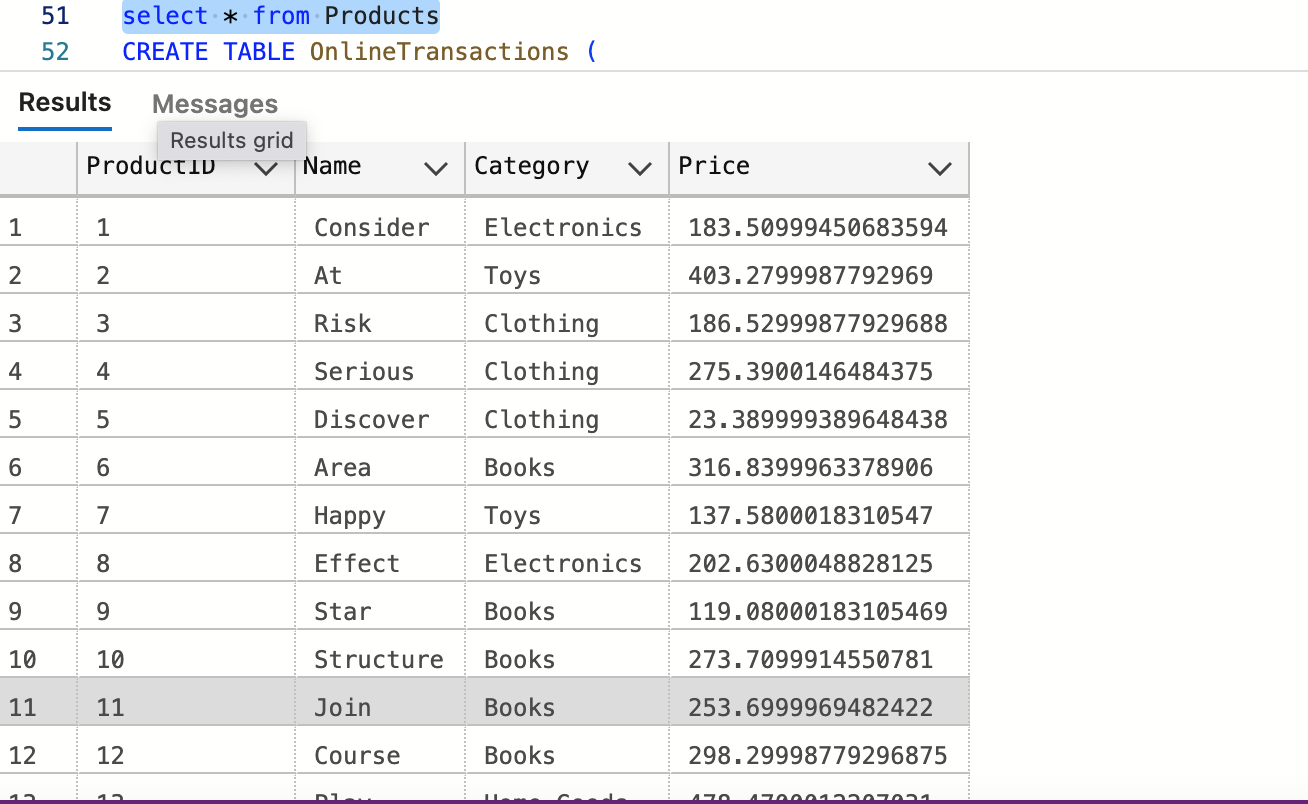
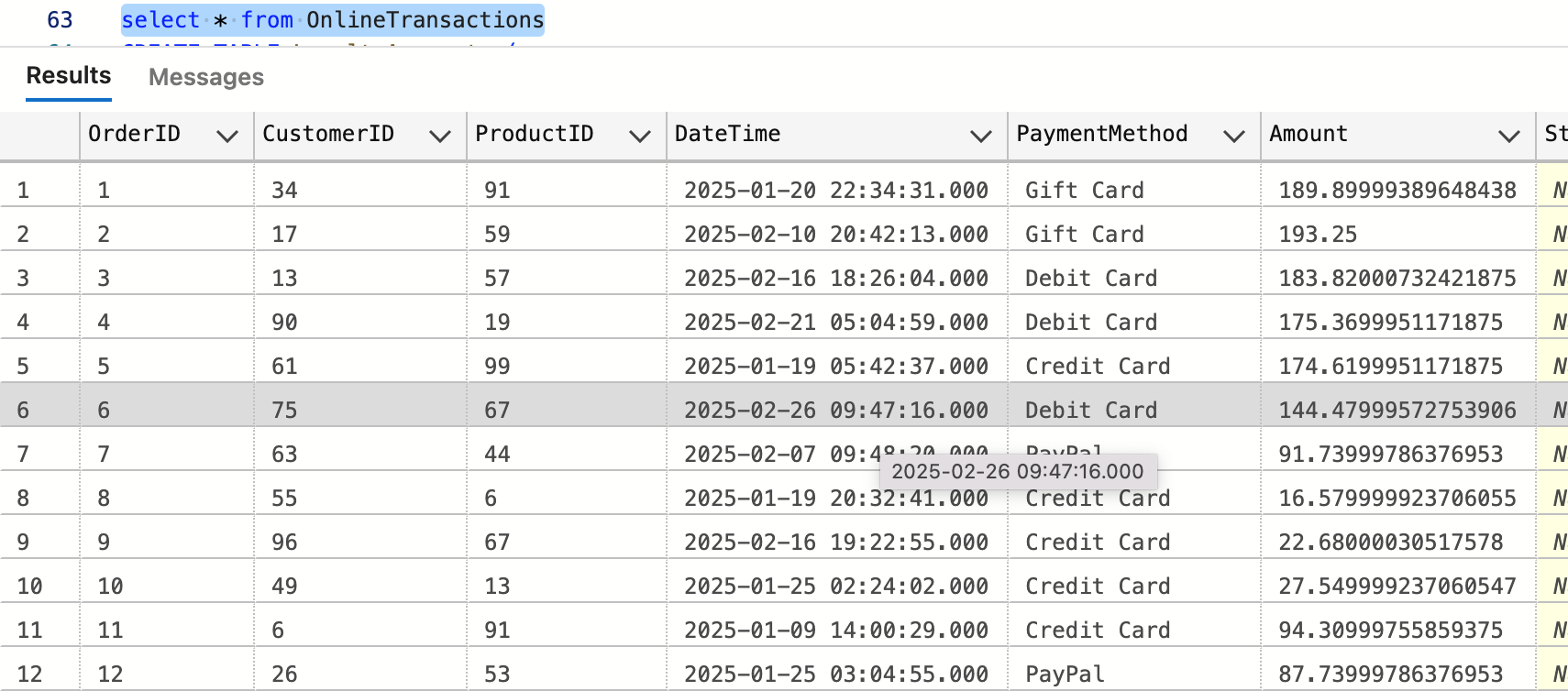
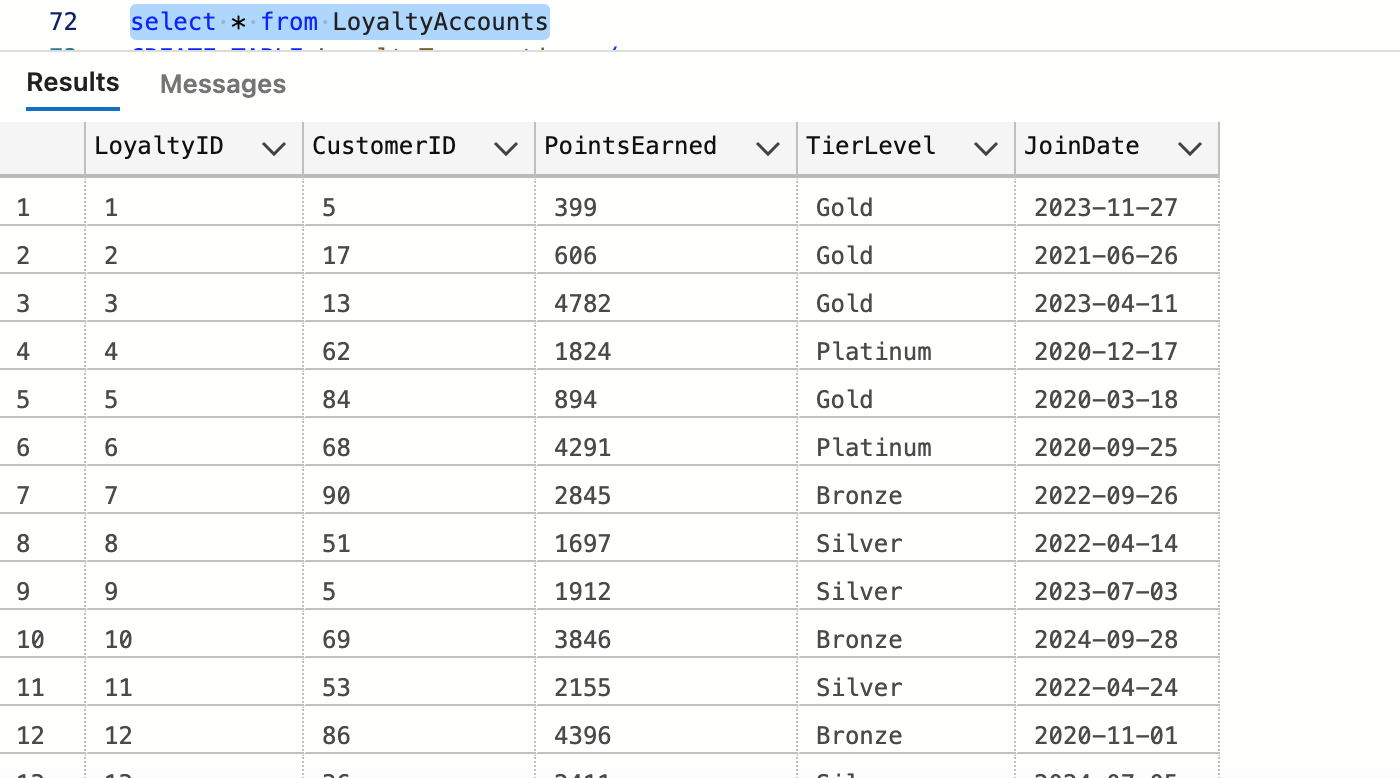
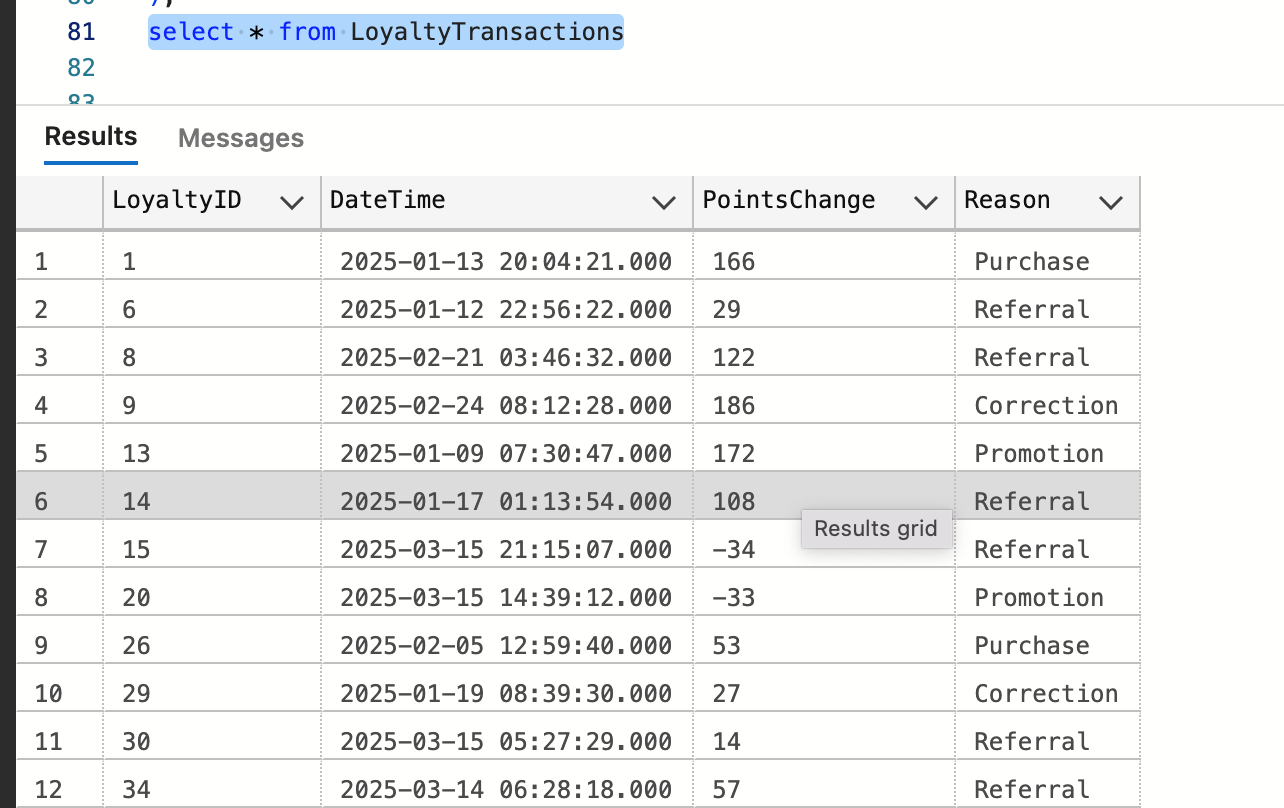
* **Purpose**: This step groups the data by **AgentID** and performs aggregation to clean and consolidate the data.
* **GroupBy**: The data is grouped by **AgentID**.
* **Aggregations**:
  + **Name**: The first(Name) function ensures that for each **AgentID**, the first **Name** encountered in the group is retained.
  + **Department**: The last(Department) function ensures that the last department value for each **AgentID** is kept.
  + **Shift**: The max(Shift) function retains the maximum (most recent) **Shift** value.
  + **Total\_Agent**: The count(AgentID) counts the total number of occurrences of **AgentID** in the group.

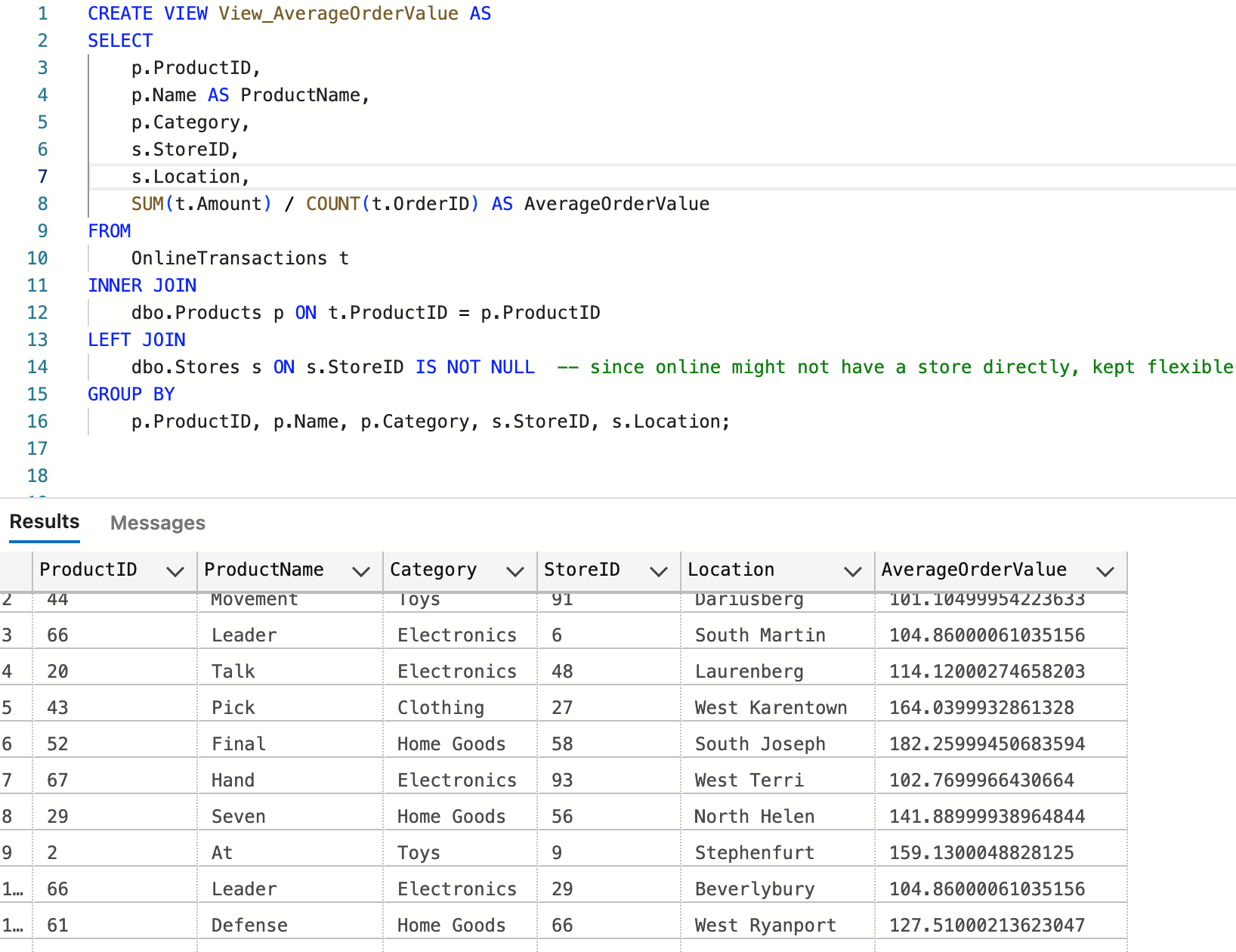
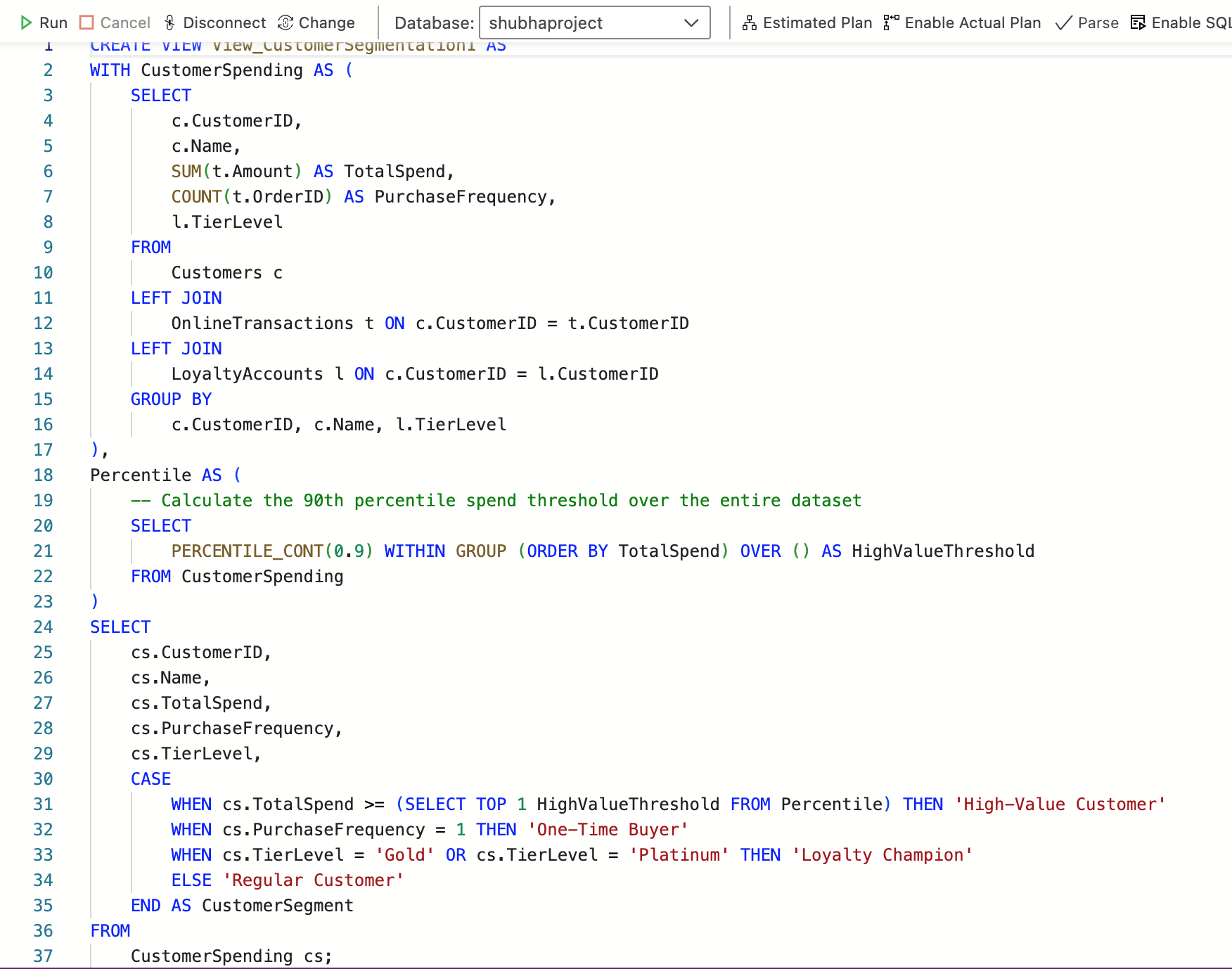
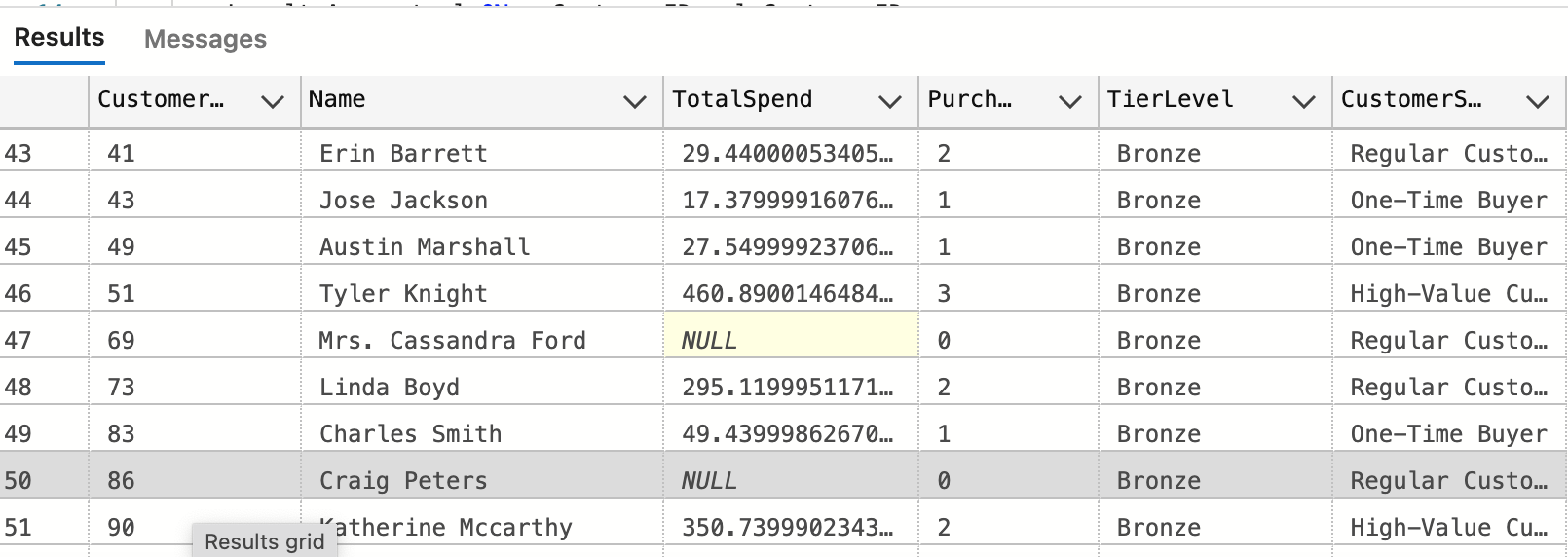
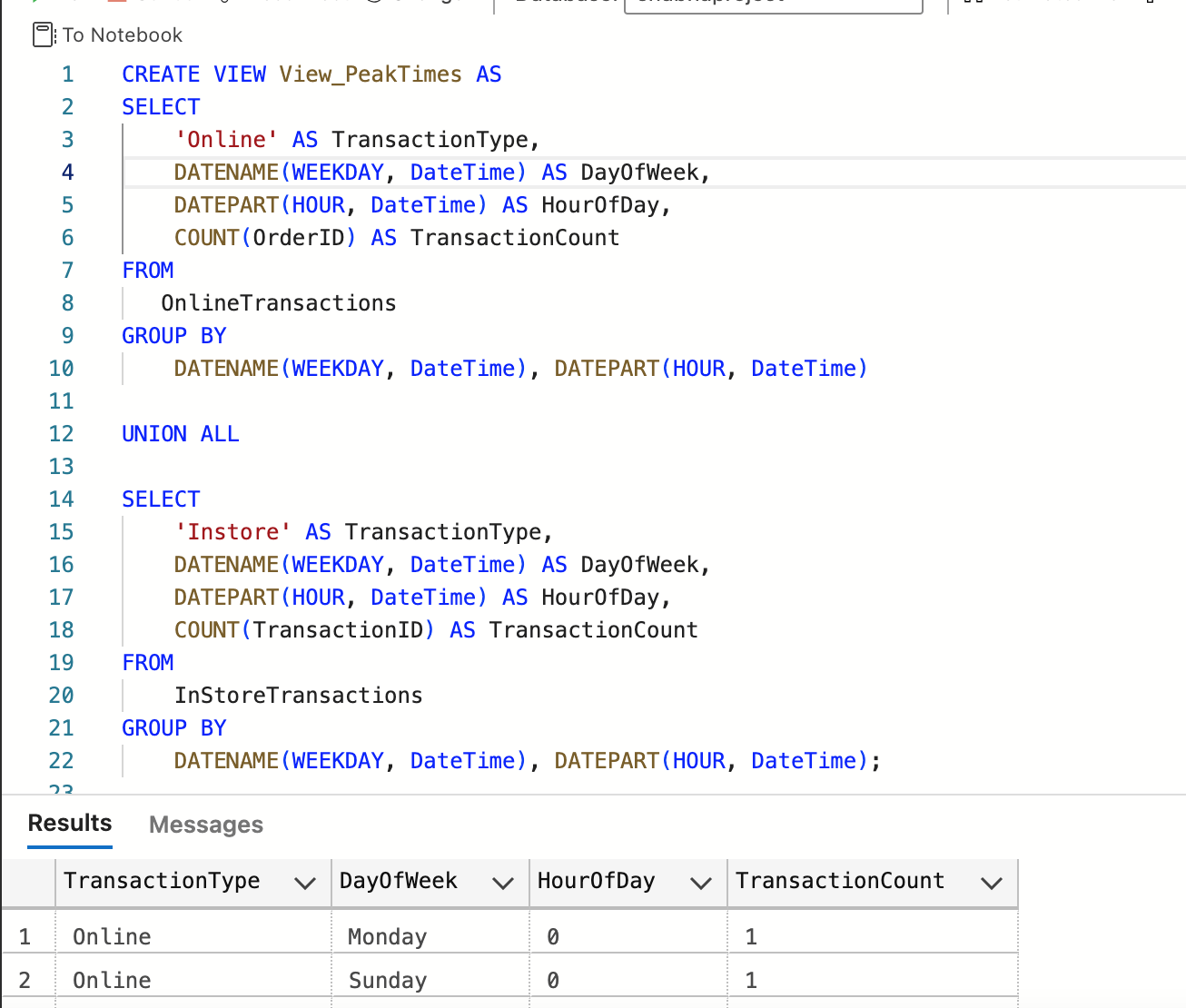
  **Filter Transformation (After Aggregate)**:

* **Purpose**: This filter ensures that only the groups where there is exactly **1** occurrence of **AgentID** are passed forward.
* **Logic**: If there are duplicates in the **AgentID** column (e.g., multiple records for the same agent), they are removed, ensuring that only records with a single **AgentID** per group are retained.

**Sink Transformation**:

* **Purpose**: After all the transformations are applied, the data is written to the target SQL Server table.
* **Linked Service**: The sink is connected to an Azure SQL Database using the **AzureSqlDatabase1** linked service.

   Data got successfully loaded in SQL tables         Step 4: Creating Views

* View 1 - for **Average Order Value (AOV)**
* SUM(Amount) / COUNT(OrderID) per product, category, and location.
*  View 2 - for Segment customers based on total spend, purchase frequency, and loyalty tier (**LoyaltyAccounts.TierLevel**).
  + Example: "High-Value Customers" (Top 10% spenders), "One-Time Buyers," "Loyalty Champions."
*   View 3 - for Analyze **DateTime** to find peak days and times in-store vs. online. 

View 4 - for Number of interactions and resolution success rates per agent (**ResolutionStatus**).

* 