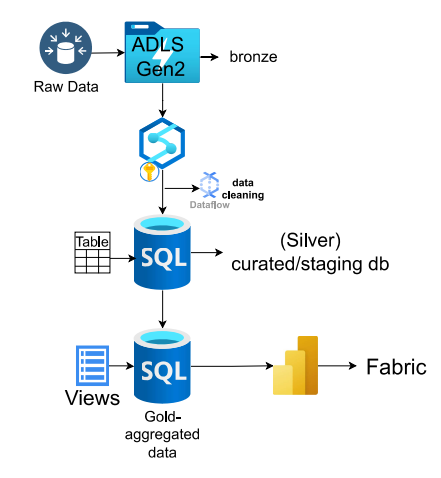
**Customer 360 View Data Integration**

**Project - 3**

**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**

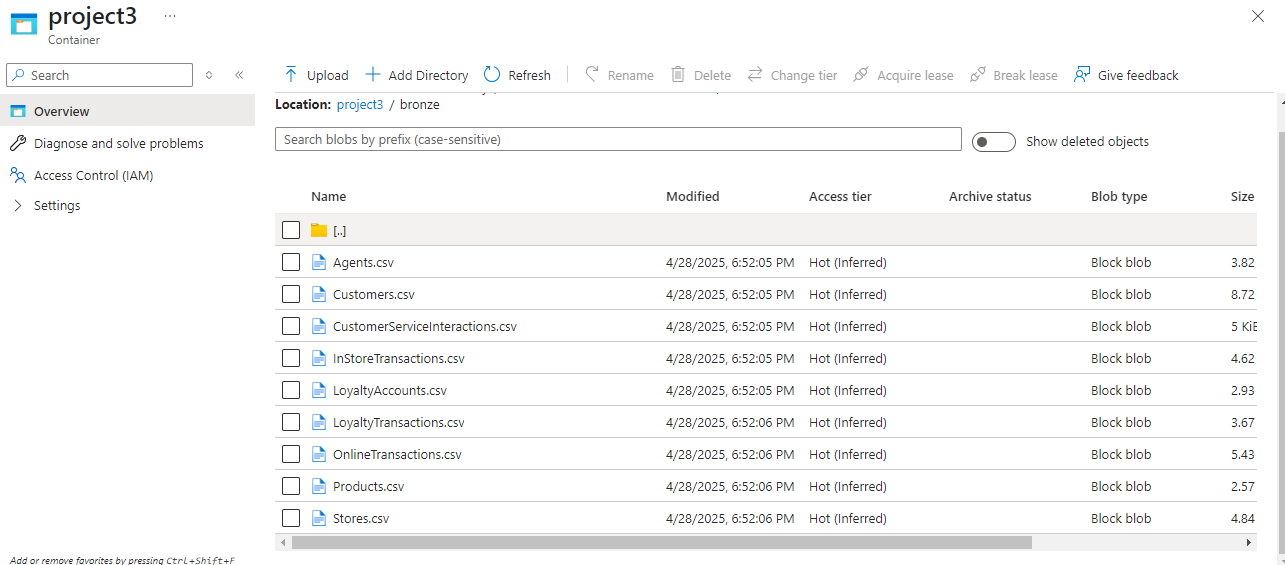


### **Tools and Technologies**

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

**Ingesting Data from multiple sources to Azure Data Lake Storage Gen2(bronze).**

* Manually ingested multiples files into the bronze layer.



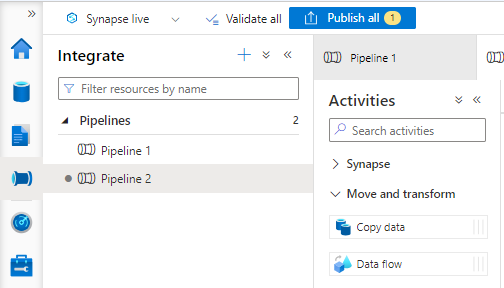
* Transforming the data in bronze and loading into silver layer.
* Before performing the transformations, create tables for each file and establish relationship between the tables in Azure SQL database to store the transformed data.

**Tables Scripts:**

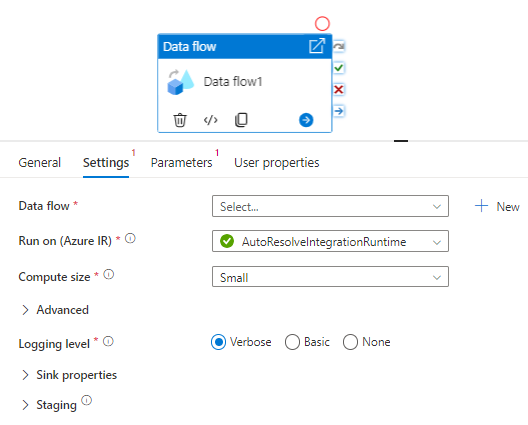
|  |
| --- |
| CREATE TABLE Customers (  CustomerID INT PRIMARY KEY,  Name VARCHAR(100),  Email VARCHAR(100),  Address VARCHAR(255) );  CREATE TABLE Products (  ProductID INT PRIMARY KEY,  Name VARCHAR(100),  Category VARCHAR(50),  Price DECIMAL(10, 2) );  CREATE TABLE OnlineTransactions (  OrderID INT PRIMARY KEY,  CustomerID INT,  ProductID INT,  DateTime DATETIME,  PaymentMethod VARCHAR(50),  Amount DECIMAL(10, 2),  Status VARCHAR(20),  FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID),  FOREIGN KEY (ProductID) REFERENCES Products(ProductID) );  CREATE TABLE Stores (  StoreID INT PRIMARY KEY,  Location VARCHAR(100),  Manager VARCHAR(100),  OpenHours VARCHAR(50) );  CREATE TABLE InStoreTransactions (  TransactionID INT PRIMARY KEY,  CustomerID INT,  StoreID INT,  DateTime DATETIME,  Amount DECIMAL(10, 2),  PaymentMethod VARCHAR(50),  FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID),  FOREIGN KEY (StoreID) REFERENCES Stores(StoreID) );  CREATE TABLE Agents (  AgentID INT PRIMARY KEY,  Name VARCHAR(100),  Department VARCHAR(50),  Shift VARCHAR(50) );  CREATE TABLE CustomerServiceInteractions (  InteractionID INT PRIMARY KEY,  CustomerID INT,  DateTime DATETIME,  AgentID INT,  IssueType VARCHAR(50),  ResolutionStatus VARCHAR(50),  FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID),  FOREIGN KEY (AgentID) REFERENCES Agents(AgentID) );  CREATE TABLE LoyaltyAccounts (  LoyaltyID INT PRIMARY KEY,  CustomerID INT,  PointsEarned INT,  TierLevel VARCHAR(20),  JoinDate DATE,  FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID) );  CREATE TABLE LoyaltyTransactions (  LoyaltyID INT,  DateTime DATETIME,  PointsChange INT,  Reason VARCHAR(100),  PRIMARY KEY (LoyaltyID, DateTime),  FOREIGN KEY (LoyaltyID) REFERENCES LoyaltyAccounts(LoyaltyID) ); |

**Transformations in Azure Synapse analytics & storing the data in Azure SQL DB.**

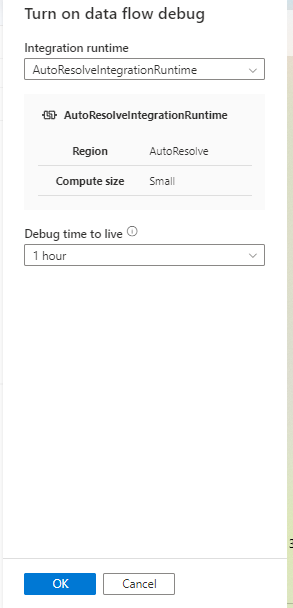
* Open synapse studio.
* Create a pipeline from integrate tab
* Drag Dataflow from Activities under move and transform

****

* Dataflow Settings🡪+New data flow

****

* Enable Data flow debug, a dialog box appears🡪ok

****

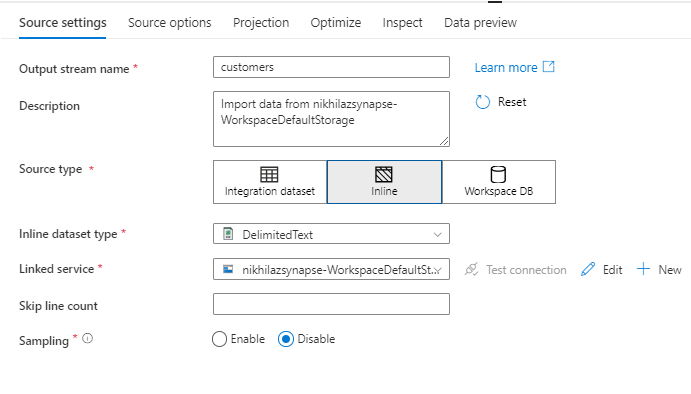
* Add Source,

1. **Source settings**,

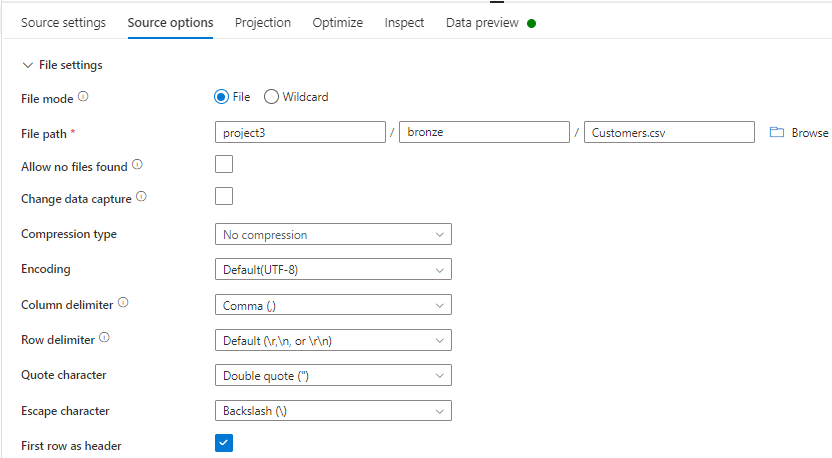
**🡪**choose source type as Inline,

**🡪**since source files are in CSV format select delimitedTex from the Inline dataset type

**🡪**Linked service as default one from the drop-down.



1. **Source Options**,

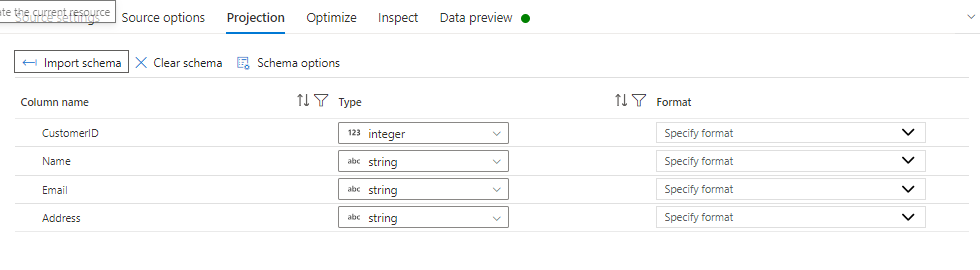


🡪provide the file path by Browse.

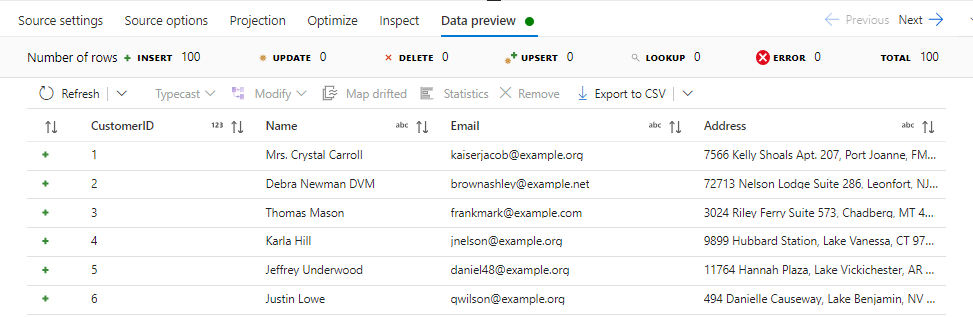
🡪check the First row as header.

1. **Projection Tab**

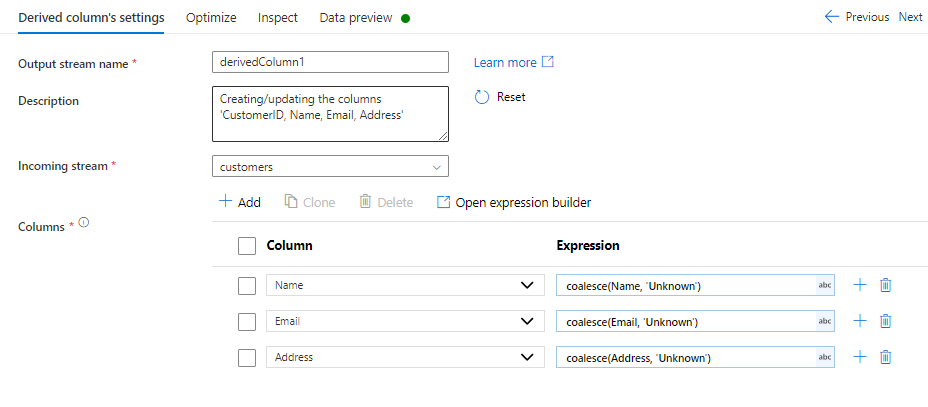
🡪import the schema



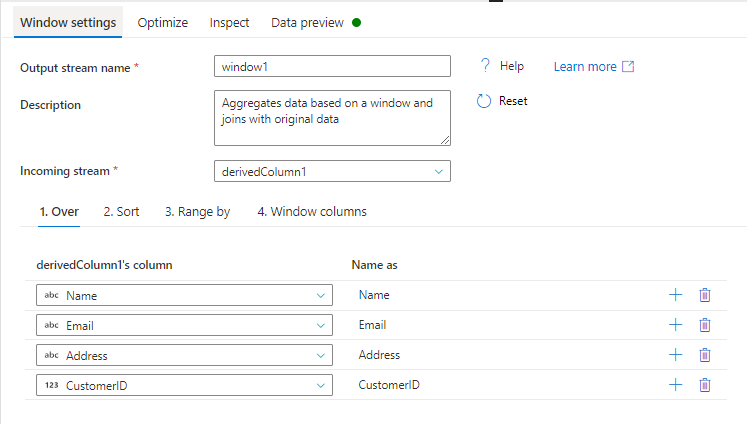
1. **Data Preview**

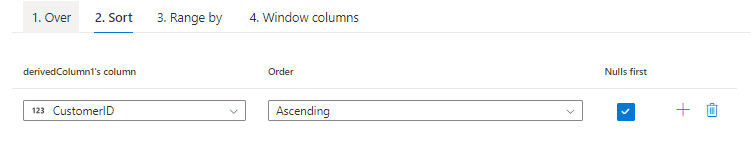
****

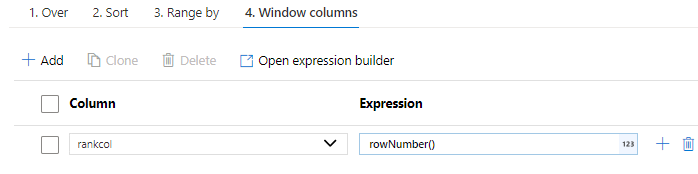
* **Add Derived column :** To replace the **Nulls** with **Coalesce()**.

****

* **Data Preview:** To see, if the transformation is applied.
* **Add Window Transformation:** To filter duplicates

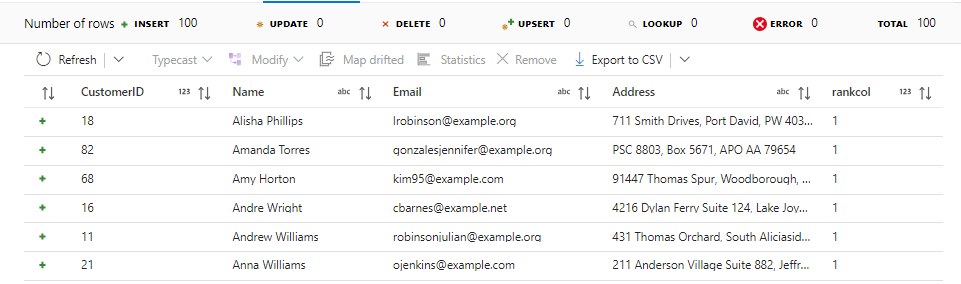
****

****

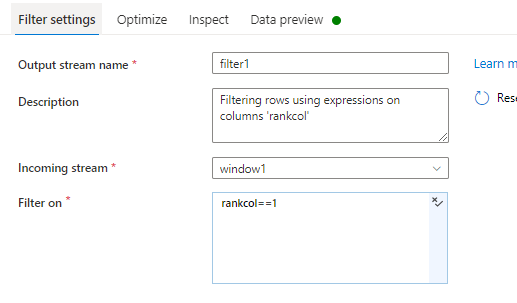
****

The **rowNumber**() function is commonly used to identify and remove duplicate rows. It assigns a unique row number within a partition

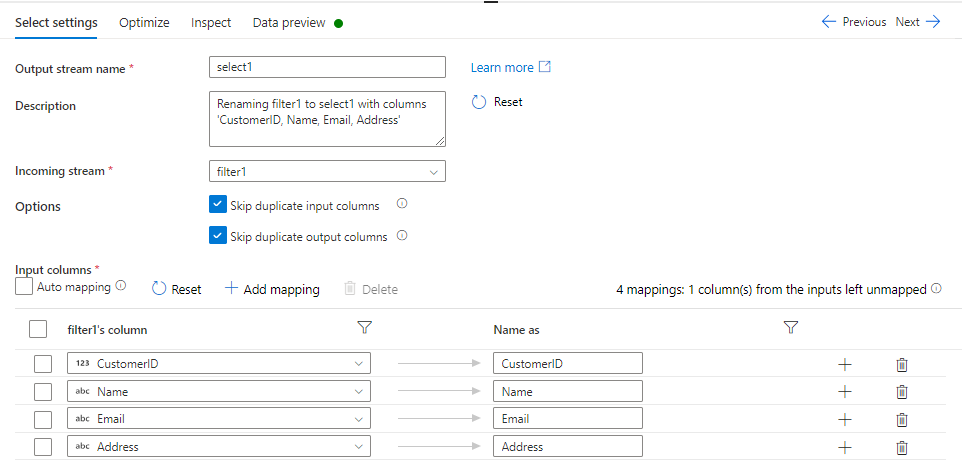
* **Data Preview**

****

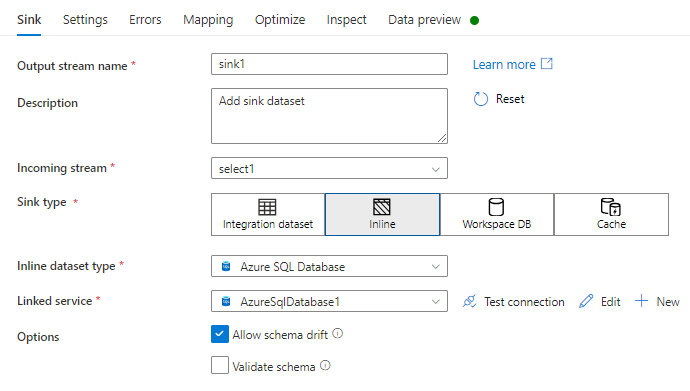
* **Add Filter Transformation**

****

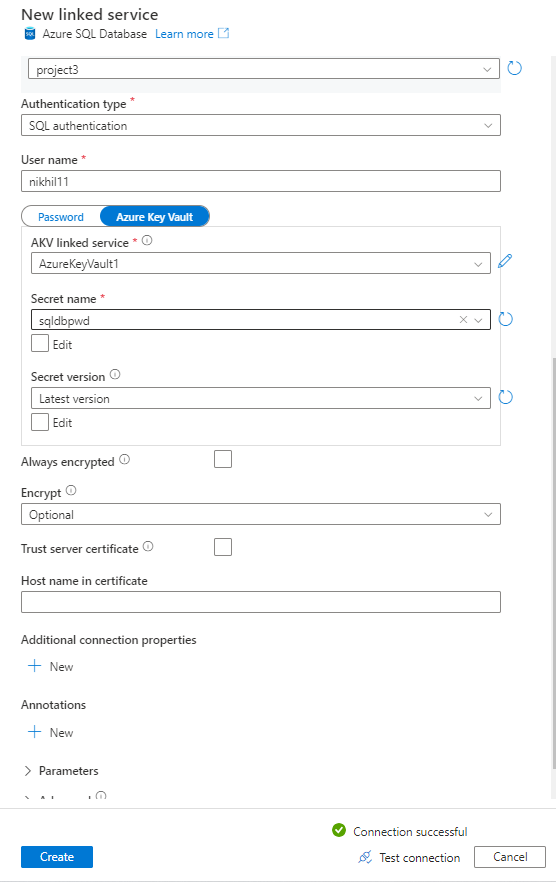
* **Add Select transformation(**Optional**)** : To unselect the rankcol

****

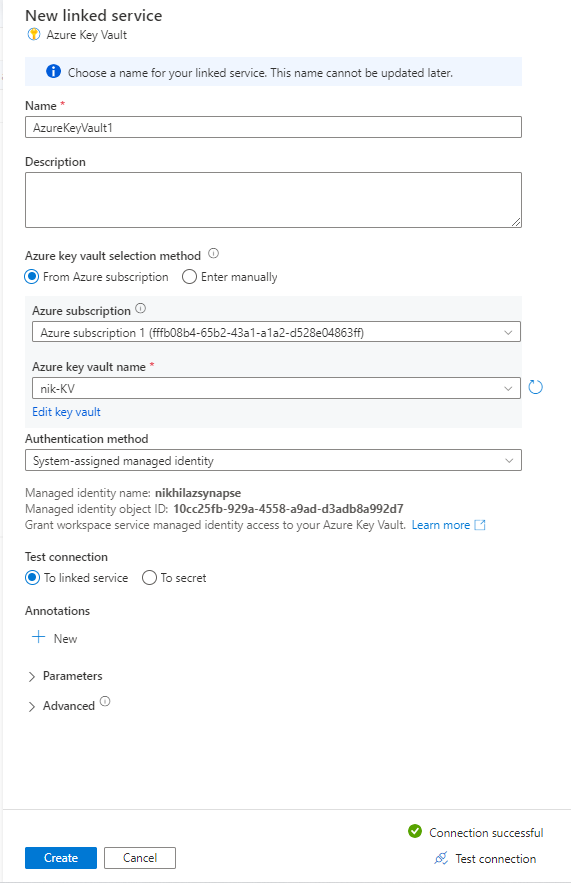
* **Add Sink,**

****

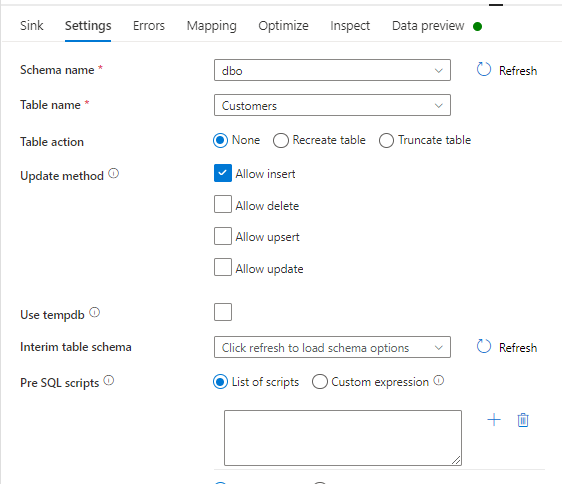
1. Choose sink type as Inline
2. Select Azure SQL Database as Inline dataset type
3. Linked service: create a new Linked service for Azure SQL DB



1. Create a Azure key Vault linked service.



1. **Sink Settings,**

****

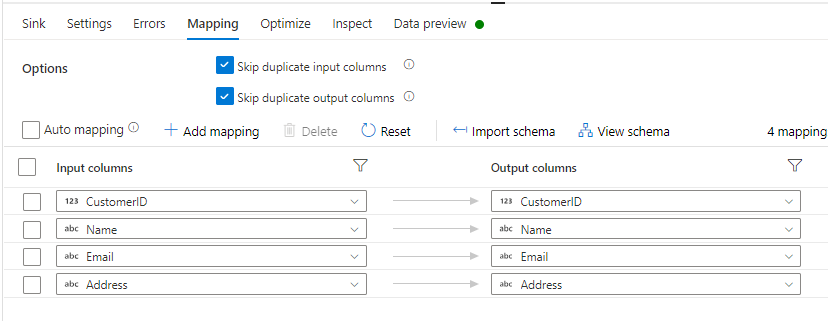
**🡪** Select the Schema

🡪Select the Table name

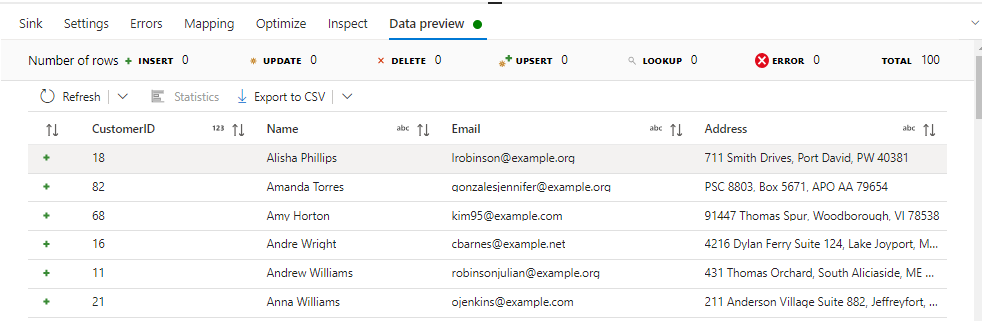
🡪use tempdb unchecked.

1. **Mapping**

🡪uncheck Auto mapping🡪import schema🡪reset.

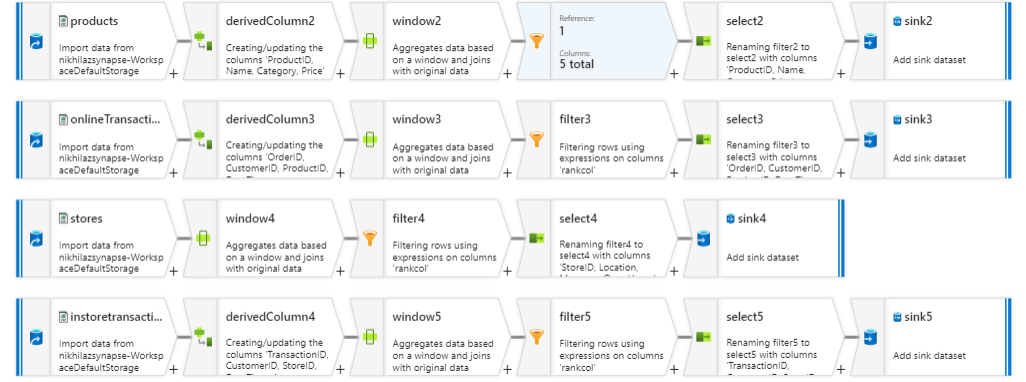


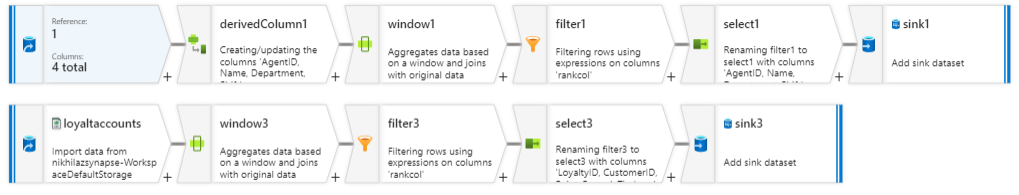
1. **Data Preview.**

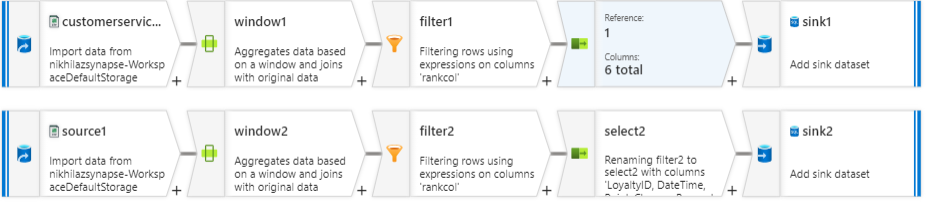
****

**Overview of the Stream**

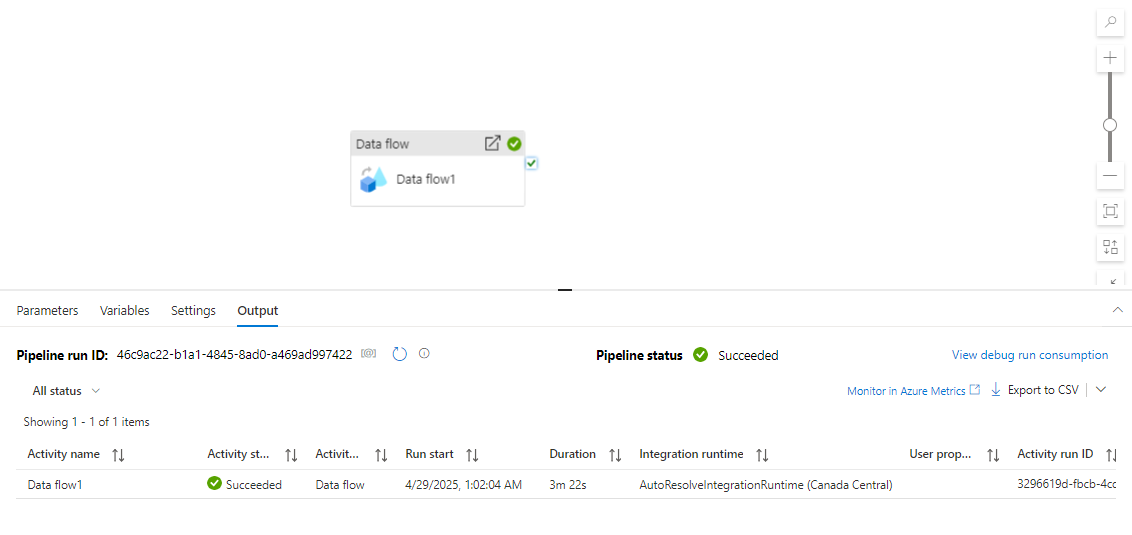
****

****

****

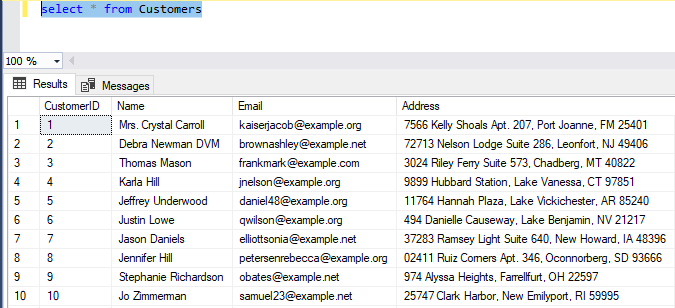
****

**Debug & Run.**

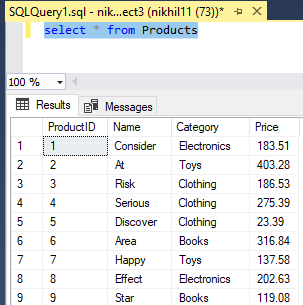


**Output in Azure SQL DB:**

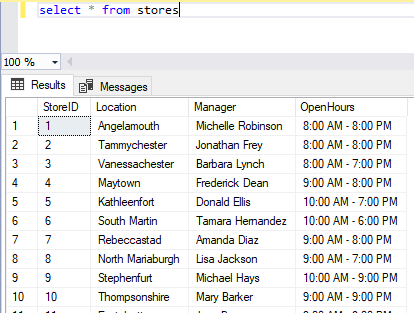
**Customers**

****

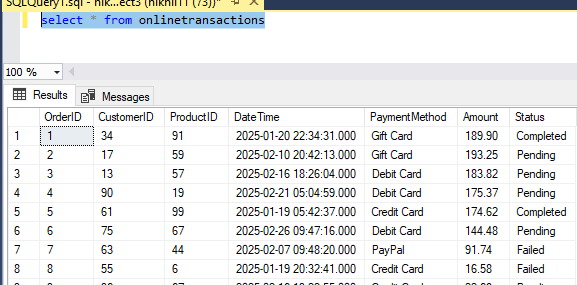
**Products**

****

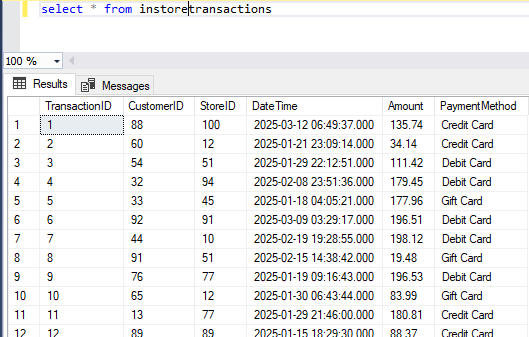
**Stores**

****

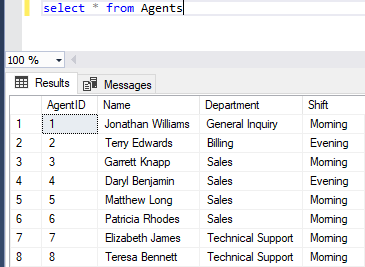
**OnlineTransactions**

****

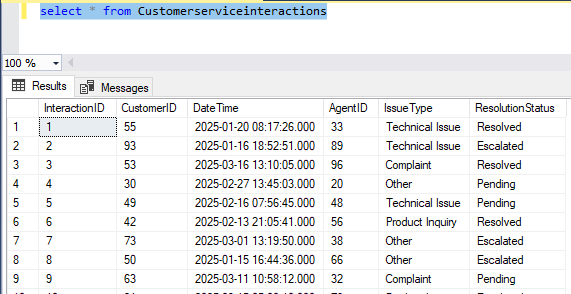
**InStoreTransactions**

****

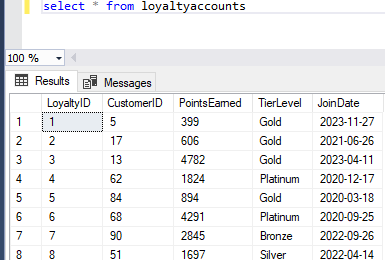
**Agents**

****

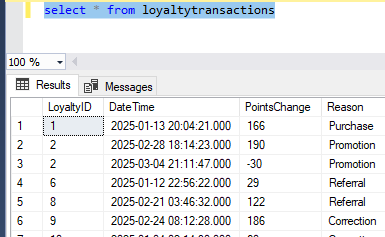
**CustomerServiceInteractions**

****

**Loyaltyaccounts**

****

**LoyaltyTransactions**

****

**Creating views in Azure SQL DB(GOLD).**

**VIEW-1:**

**Average Order Value (AOV):** SUM(Amount) / COUNT(OrderID) per product, category, and location

CREATE VIEW View\_AverageOrderValue AS

SELECT

p.ProductID,

p.Name AS ProductName,

p.Category,

s.StoreID,

s.Location,

SUM(t.Amount) / COUNT(t.OrderID) AS AverageOrderValue

FROM

onlineTransaction t

INNER JOIN

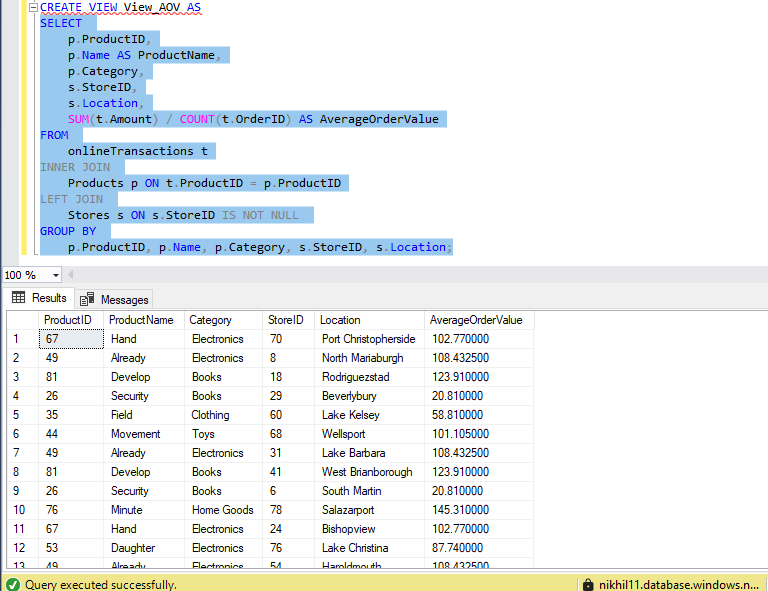
dbo.Products p ON t.ProductID = p.ProductID

LEFT JOIN

dbo.Stores s ON s.StoreID IS NOT NULL -- since online might not have a store directly, kept flexible

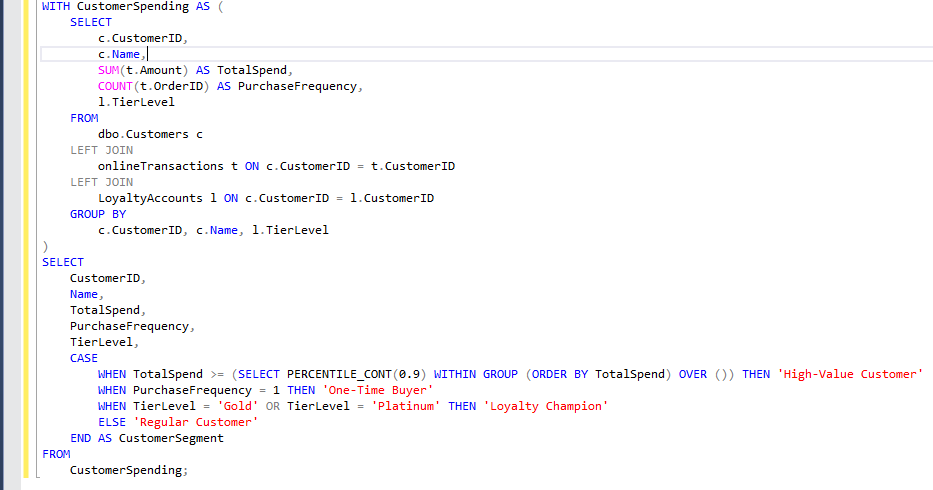
GROUP BY

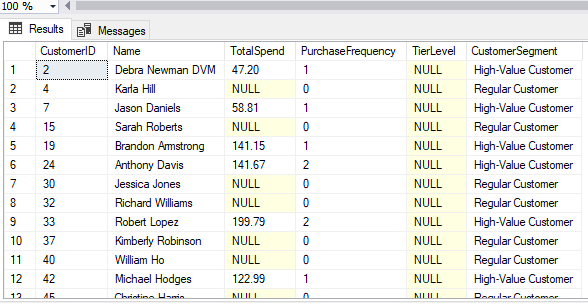
p.ProductID, p.Name, p.Category, s.StoreID, s.Location;



**VIEW-2**

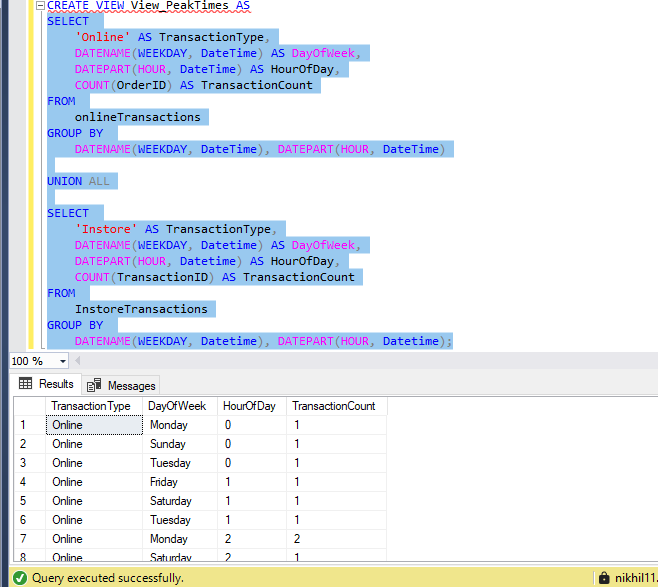
Segment customers based on total spend, purchase frequency, and loyalty tier (**LoyaltyAccounts.TierLevel**).



****

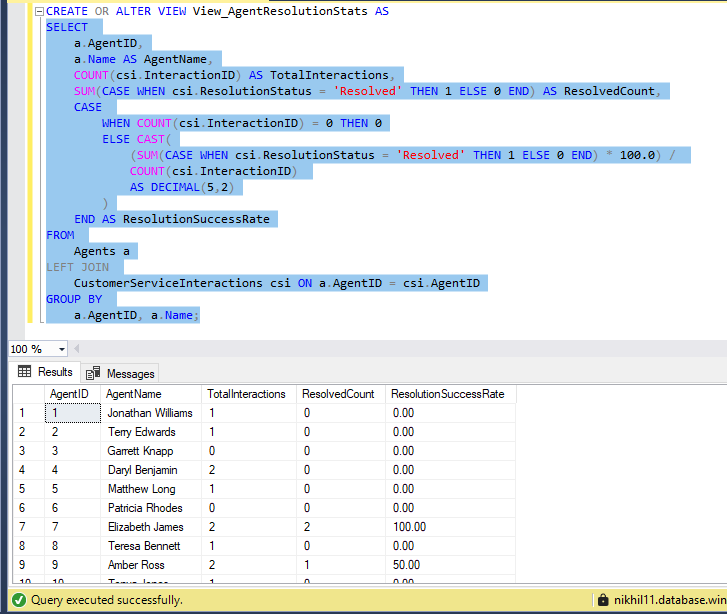
**VIEW-3**

Analyze DateTimeto find peak days and times in-store vs. online

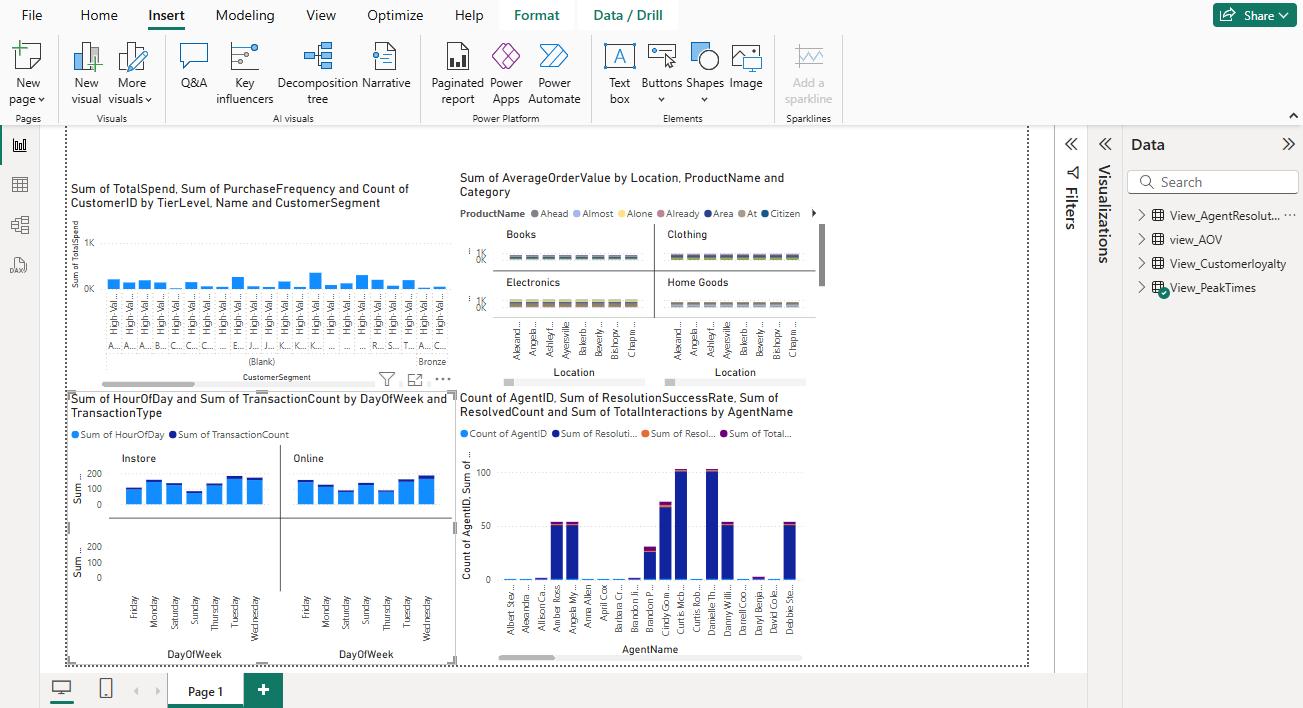


**VIEW-4**

Number of interactions and resolution success rates per agent (**ResolutionStatus**).



**PowerBI:**



**Fabric**

