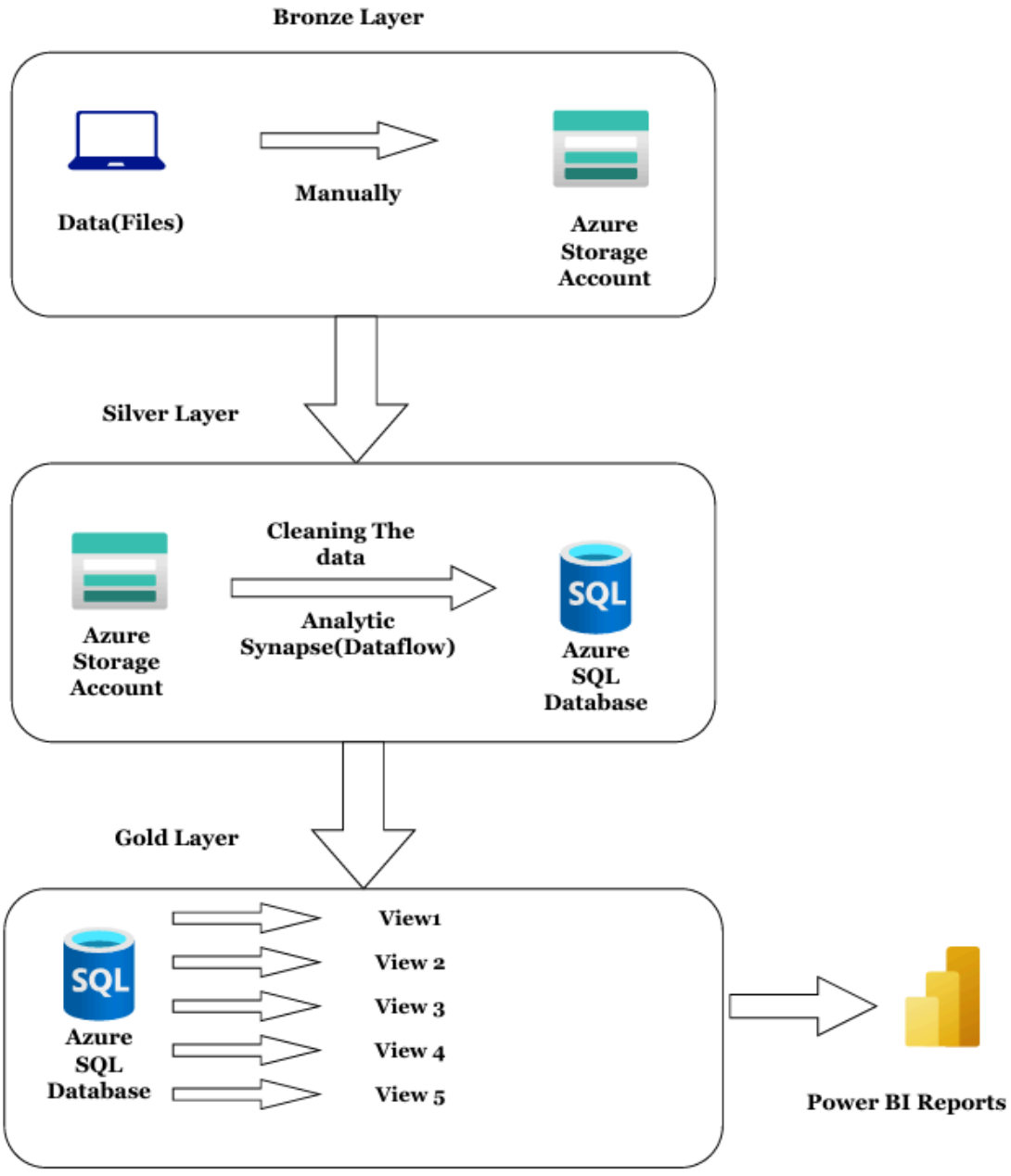
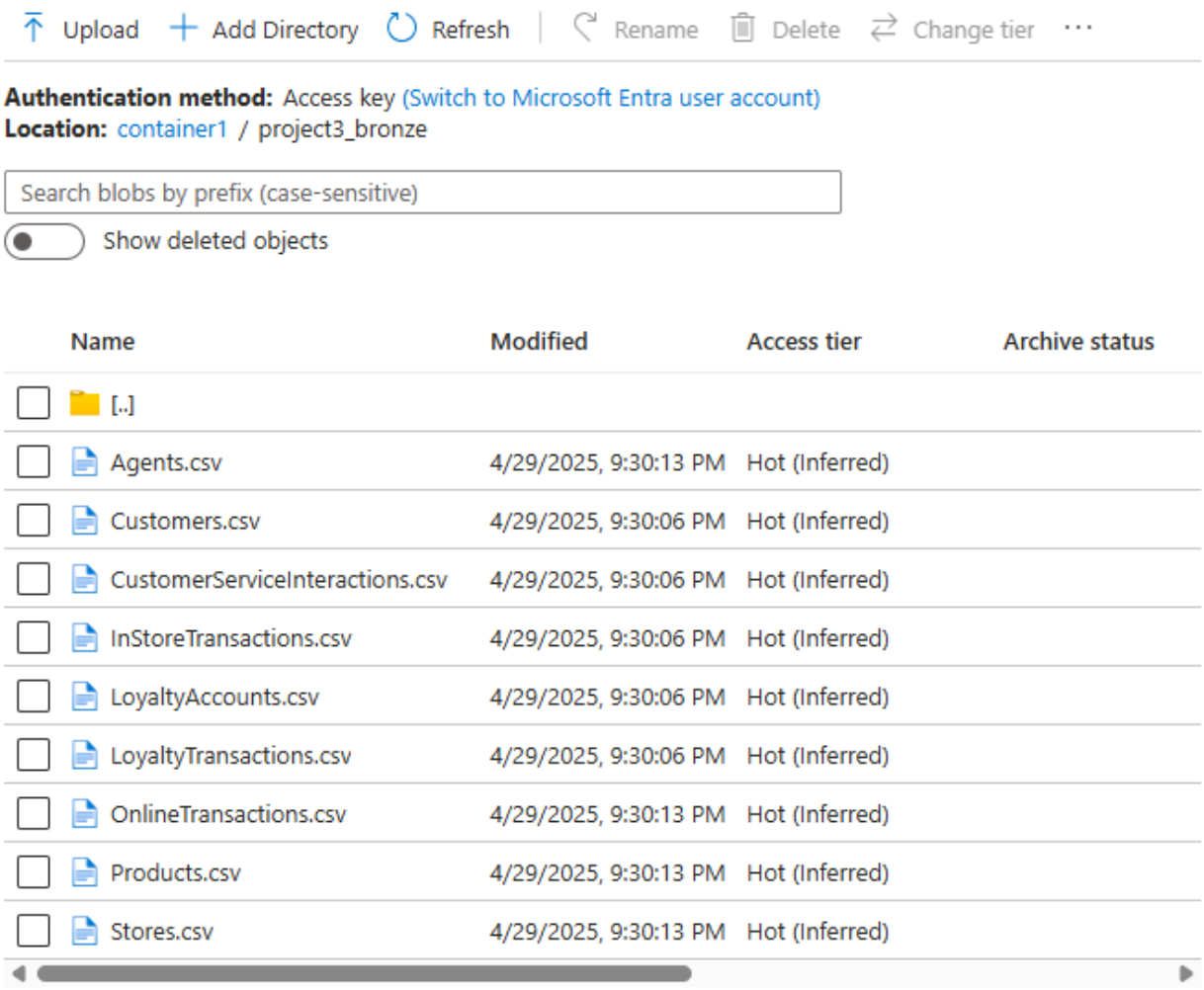
Project 3: Customer 360 Data Integration

Architecture:



Step 1: Ingest Data

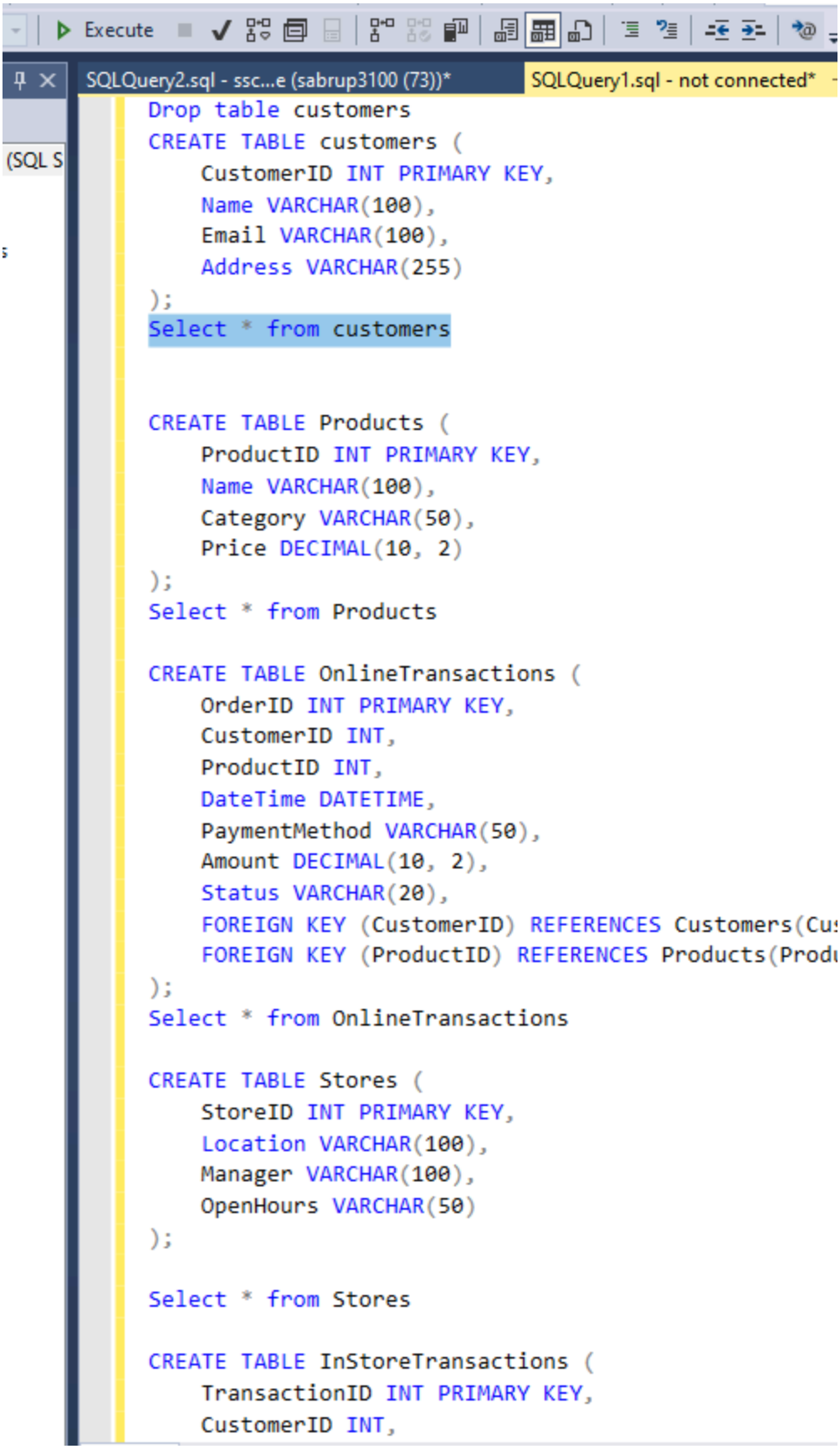
Manually uploading of all the 9 files(online, in-store, customer service, loyalty programs) into the raw container in ADLS.

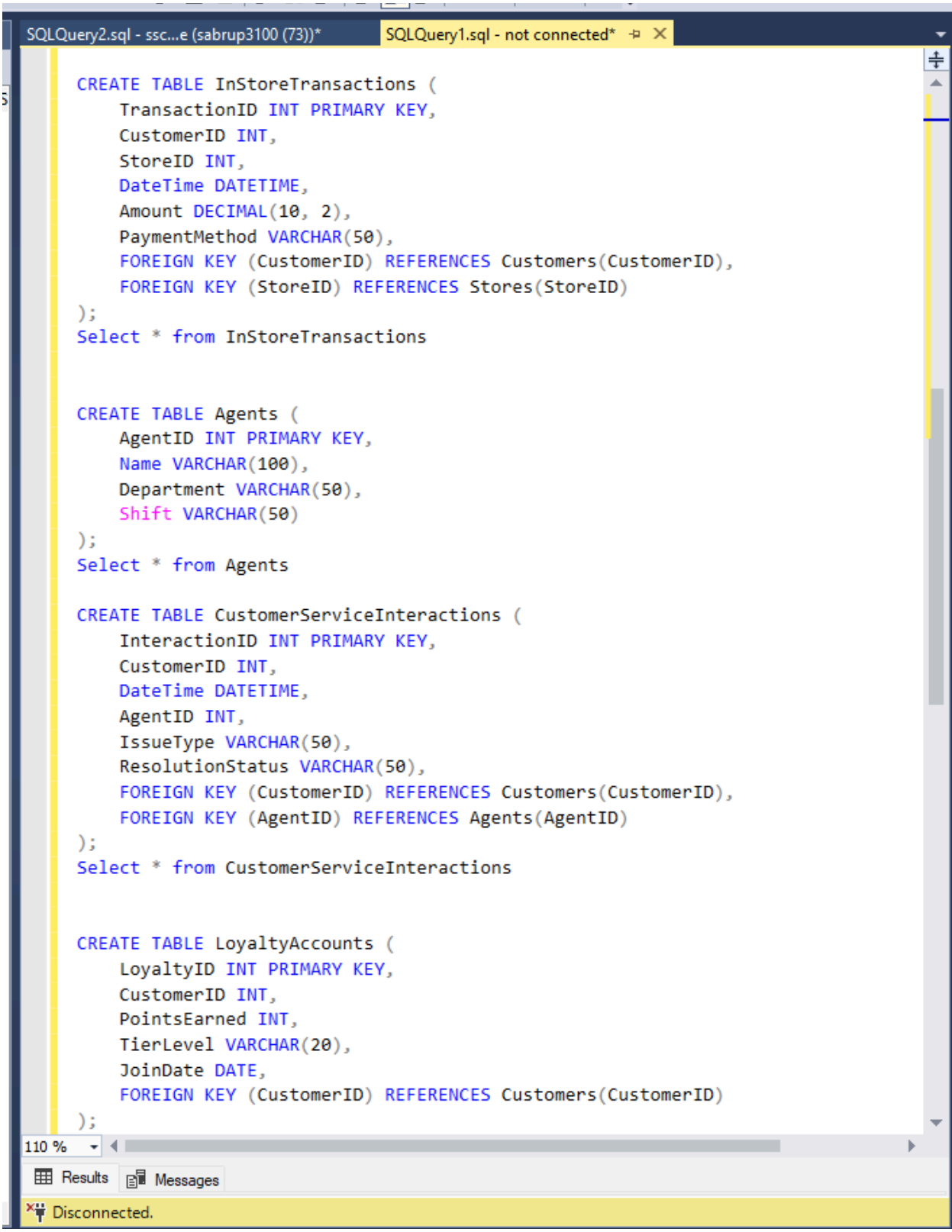


Step 2: Define Staging (Curated) Schema

- **Create DDLs for a staging (silver) database in Azure Synapse Analytics to clean and standardize the data.**

1. **First of all I have created 9 tables:**


Execute
SQLQuery2.sql - ssc...e (sabrup3100 (73))*
SQLQuery1.sql - not connected*
(SQL S
Drop table customers
CREATE TABLE customers (
CustomerID INT PRIMARY KEY,
Name VARCHAR(100),
Email VARCHAR(100),
Address VARCHAR(255)
);
Select * from customers
CREATE TABLE Products (
ProductID INT PRIMARY KEY,
Name VARCHAR(100),
Category VARCHAR(50),
Price DECIMAL(10, 2)
);
Select * from Products
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(Cus
FOREIGN KEY (ProductID) REFERENCES Products(Produ
);
Select * from OnlineTransactions
CREATE TABLE Stores (
StoreID INT PRIMARY KEY,
Location VARCHAR(100),
Manager VARCHAR(100),
OpenHours VARCHAR(50)
);
Select * from Stores
CREATE TABLE InStoreTransactions (
TransactionID INT PRIMARY KEY,
CustomerID INT,

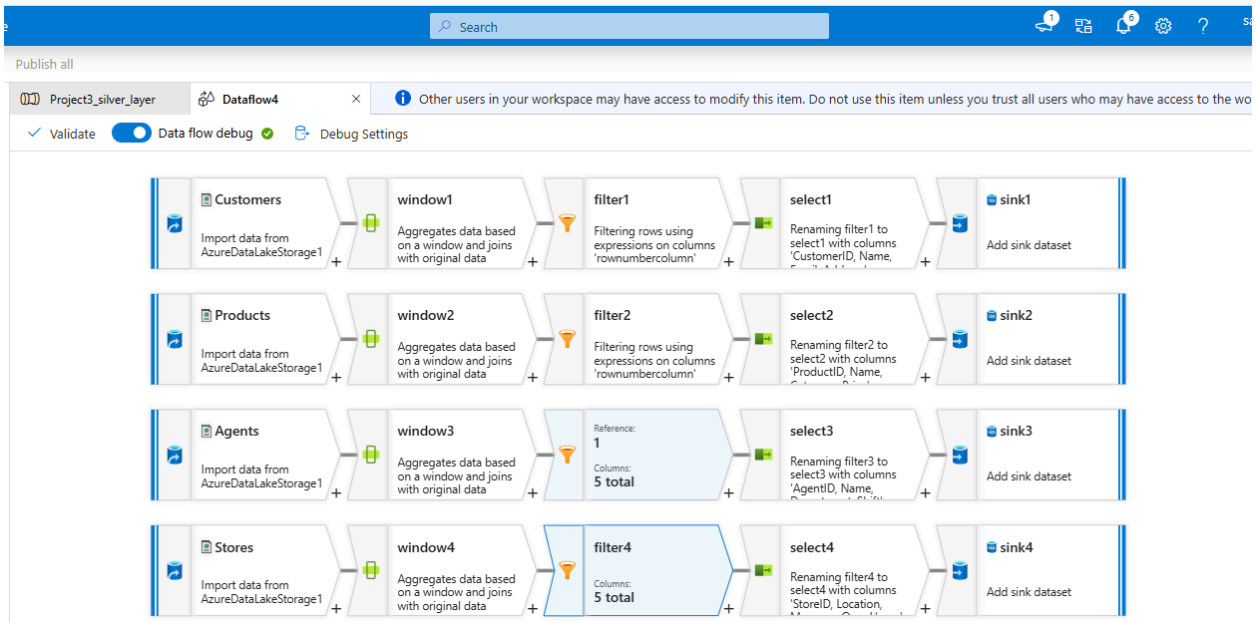

SQLQuery2.sql - ssc...e (sabrup3100 (73))*
SQLQuery1.sql - not connected*
CREATE TABLE InStoreTransactions (
TransactionID INT PRIMARY KEY,
CustomerID INT,
StoreID INT,
DateTime DATETIME,
Amount DECIMAL(10, 2),
PaymentMethod VARCHAR(50),
FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID),
FOREIGN KEY (StoreID) REFERENCES Stores(StoreID)
);
Select * from InStoreTransactions
CREATE TABLE Agents (
AgentID INT PRIMARY KEY,
Name VARCHAR(100),
Department VARCHAR(50),
Shift VARCHAR(50)
);
Select * from Agents
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)
);
Select * from CustomerServiceInteractions
CREATE TABLE LoyaltyAccounts (
LoyaltyID INT PRIMARY KEY,
CustomerID INT,
PointsEarned INT,
TierLevel VARCHAR(20),
JoinDate DATE,
FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)
);
110 %
Results
Messages
Disconnected.

```sql
CREATE TABLE LoyaltyTransactions (
    LoyaltyID INT,
    DateTime DATETIME,
    PointsChange INT,
    Reason VARCHAR(100),
    PRIMARY KEY (LoyaltyID, DateTime),
    FOREIGN KEY (LoyaltyID) REFERENCES LoyaltyAccounts(LoyaltyID)
);
Select * from LoyaltyTransactions
```

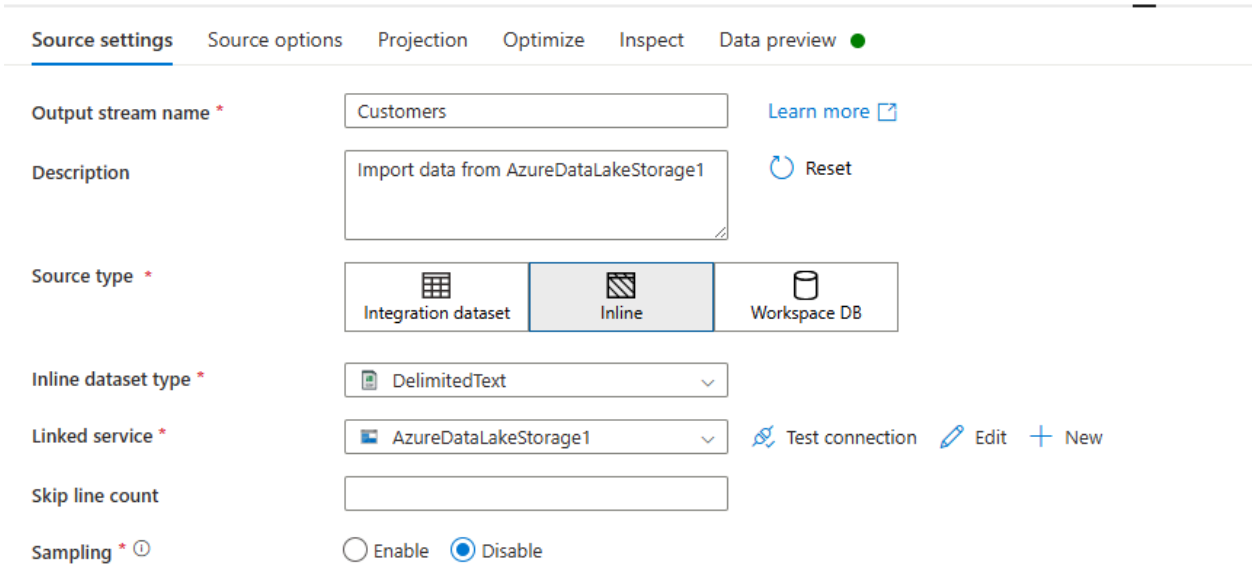
2.

For the silver layer I have created 6 Dataflows. For 4 of the files, having Primary Key and hasn't any relation with other, I have created 1 single dataflow.

The data has reference to other tables that I did separately.

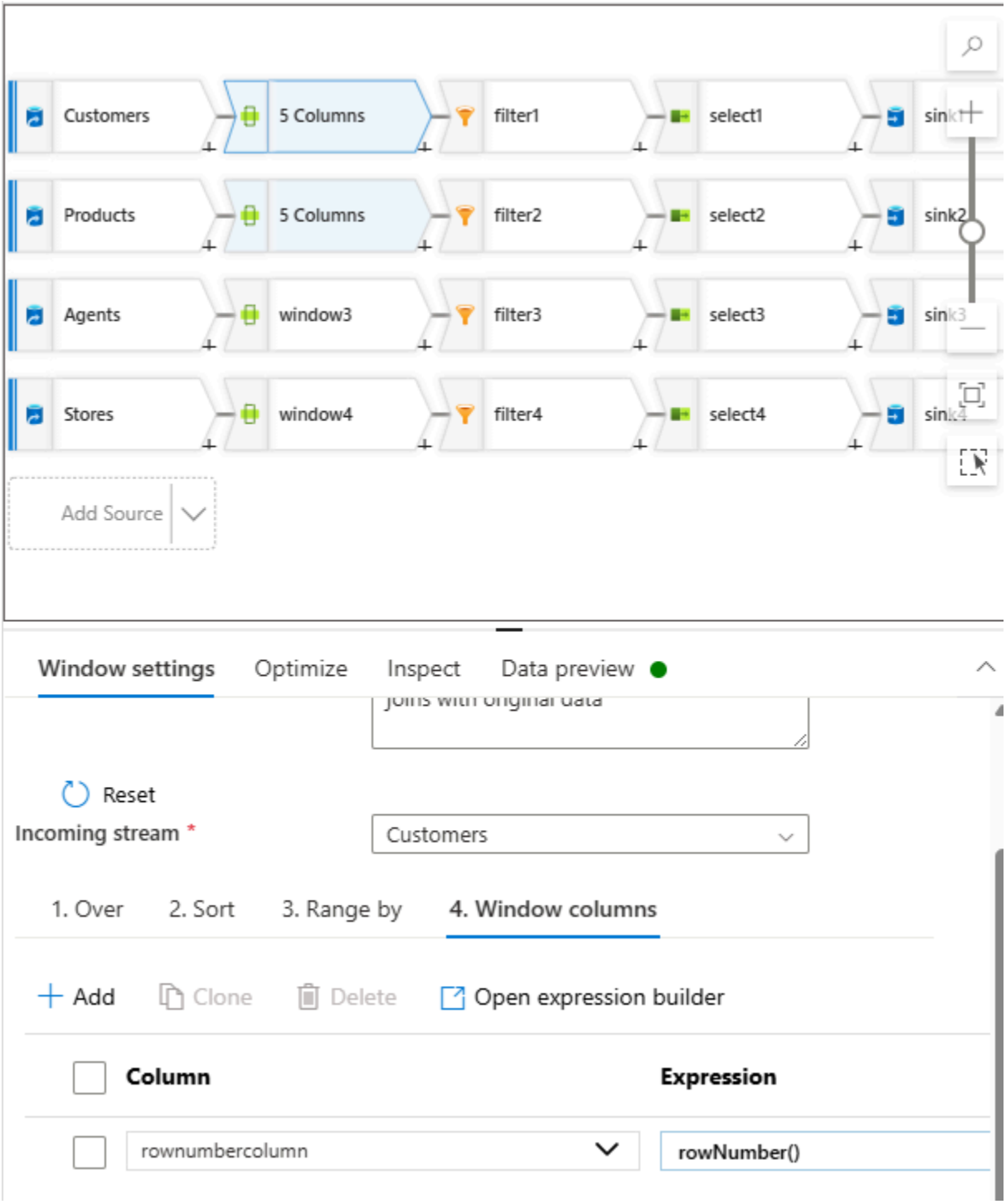


1. **In a pipeline, create a Dataflow.choose Source type, as here it's Delimited Text, datasettype and the linked service.**

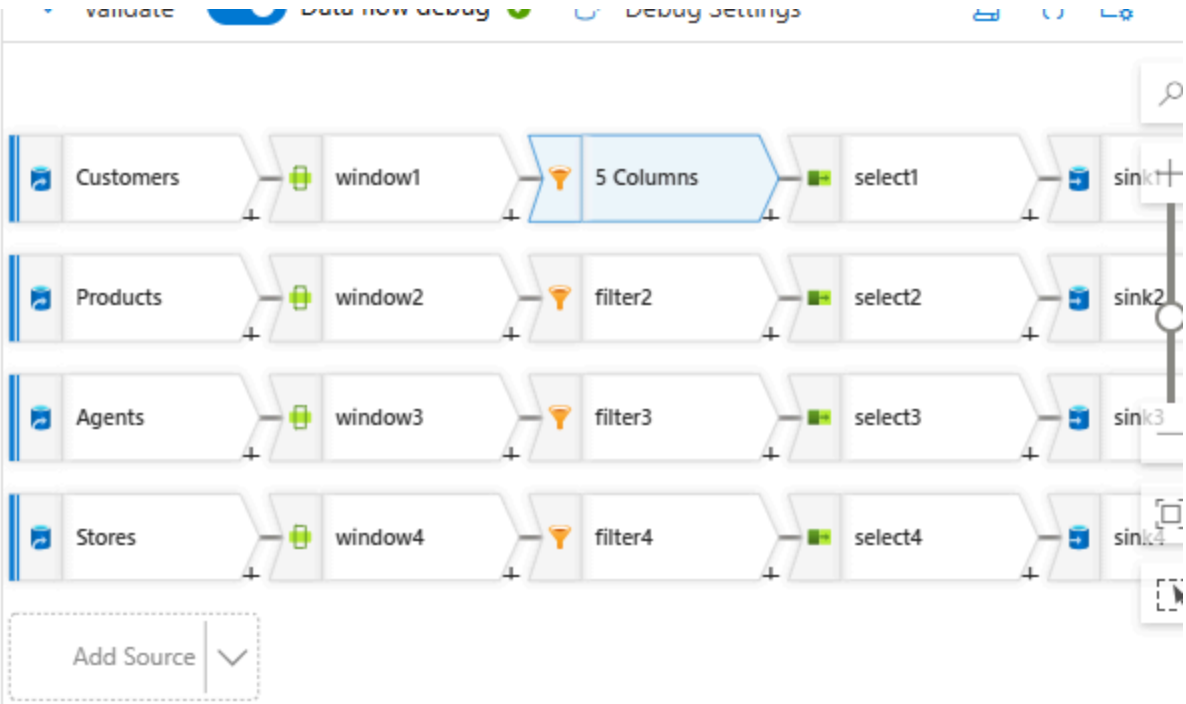


2. **Under Source option, we need to choose the path for the file.**

3. **Nextly, In window transformation I added a window column, “rownumbercolumn” which indicates the number of rows for that particular Id.**



4. **Then I used filter transformation and used filter on that rownumbercolumn. Means will process only rows having value 1.**



Filter settings | Optimize | Inspect | Data preview

Output stream name * filter1

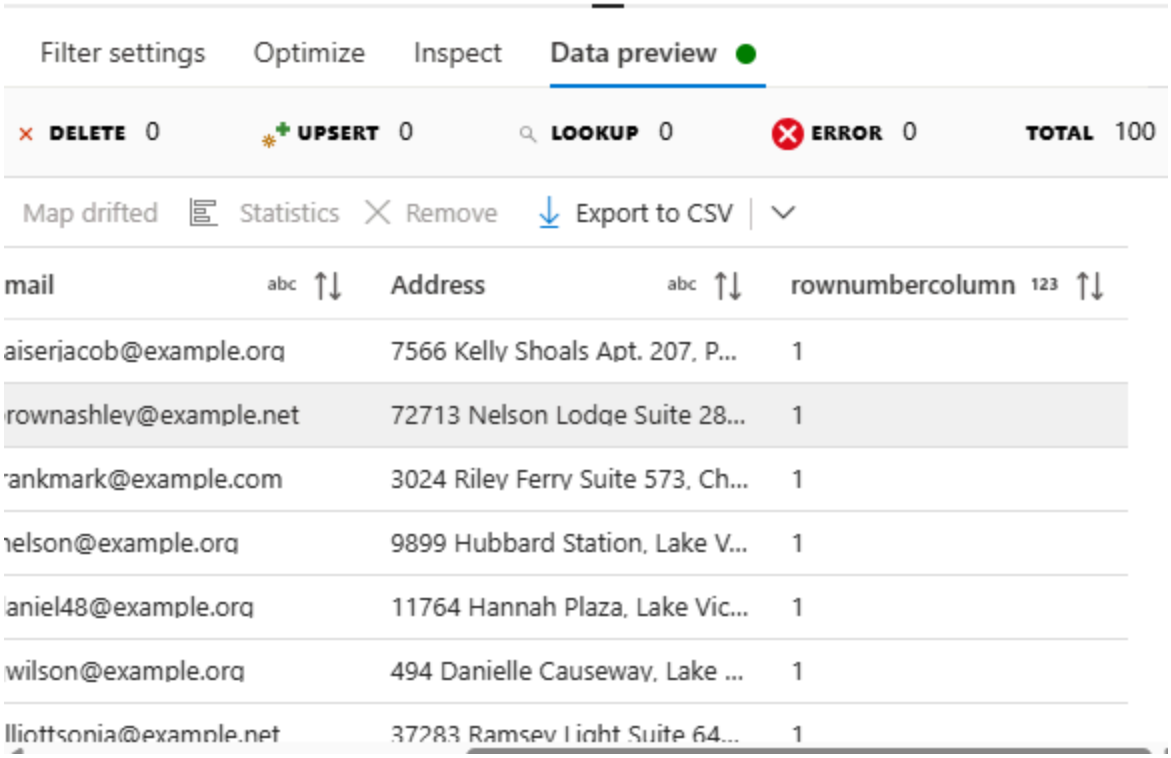
Learn more

Description: Filtering rows using expressions on columns 'rownumbercolumn'

Reset

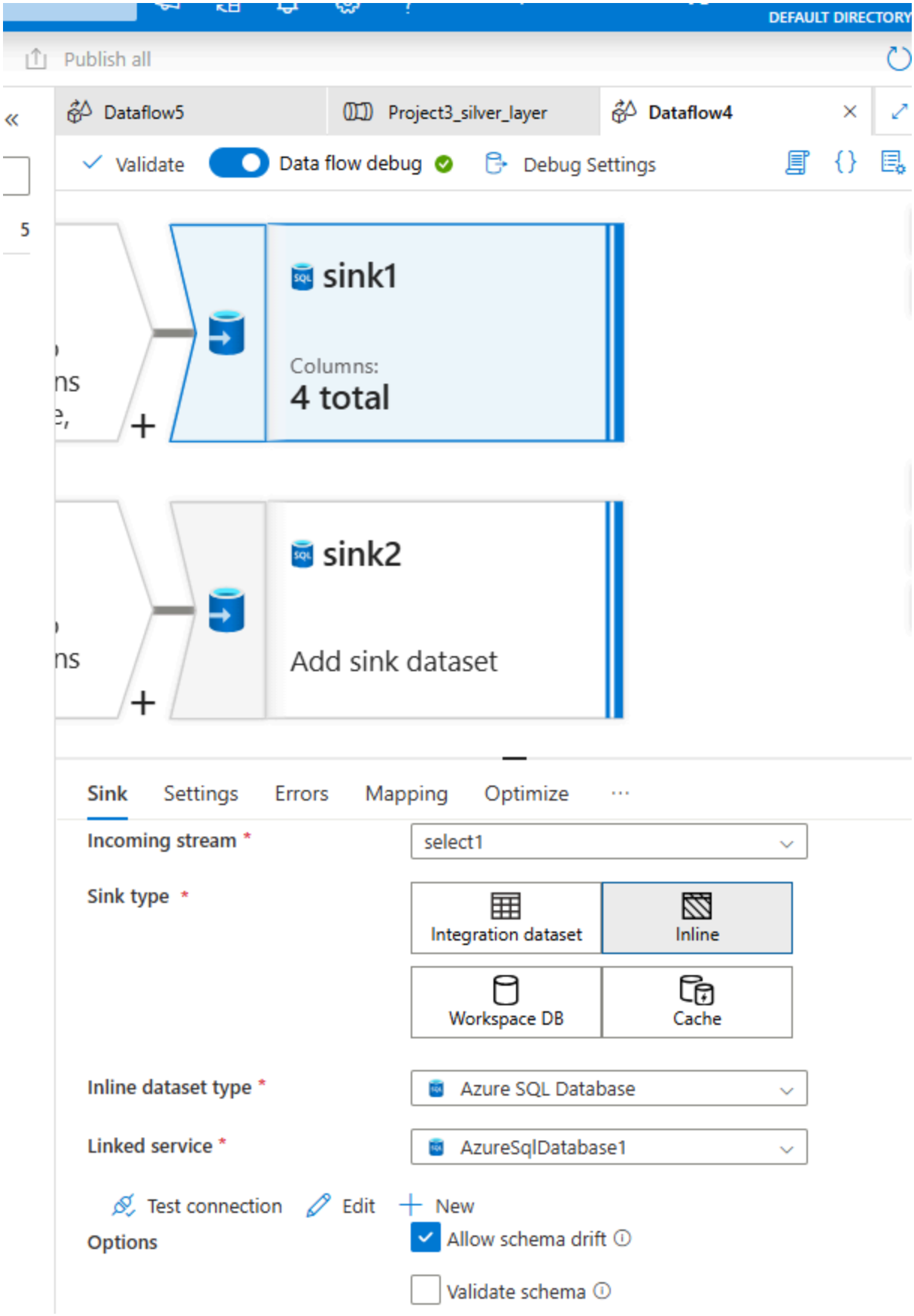
Incoming stream * window1

Filter on * rownumbercolumn == 1

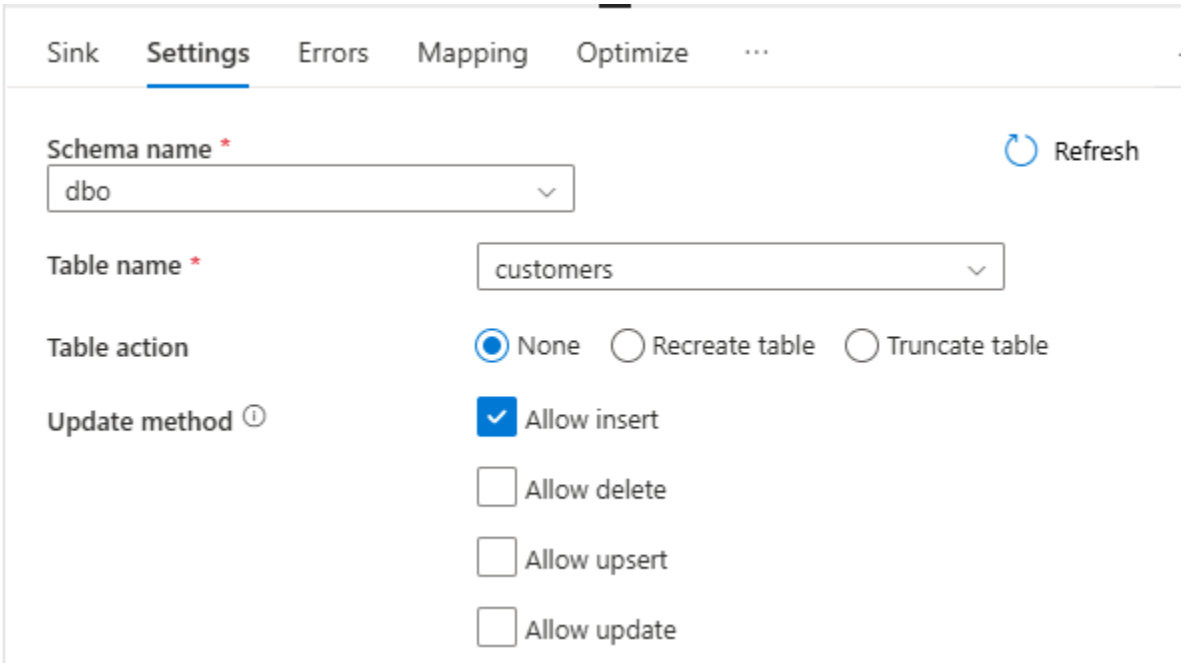


Here I am getting the rows with rownumbercolumn 1.

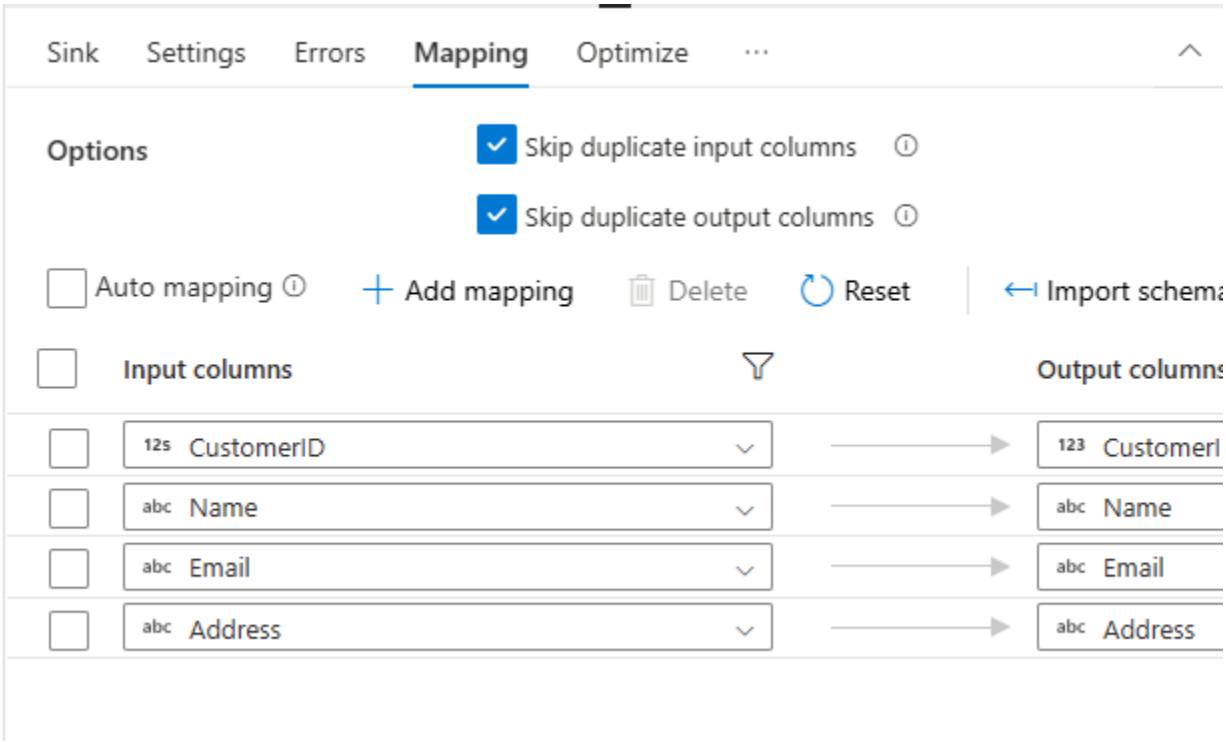
5. **Lastly, to store the data in Azure Sql tables, I used sink transformation.**


DEFAULT DIRECTORY
Publish all
Dataflow5
Project3_silver_layer
Dataflow4
Validate
Data flow debug
Debug Settings
sink1
Columns:
4 total
sink2
Add sink dataset
Sink
Settings
Errors
Mapping
Optimize
Incoming stream *
select1
Sink type *
Integration dataset
Inline
Workspace DB
Cache
Inline dataset type *
Azure SQL Database
Linked service *
AzureSqlDatabase1
Test connection
Edit
New
Options
Allow schema drift
Validate schema

With the target schema and the table:

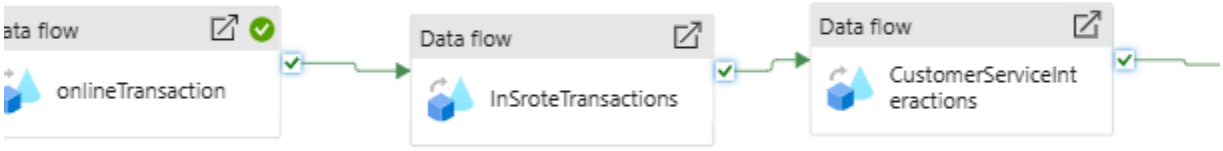


Here I removed the rownumbercolumn while mapping as it's a unnecessary column

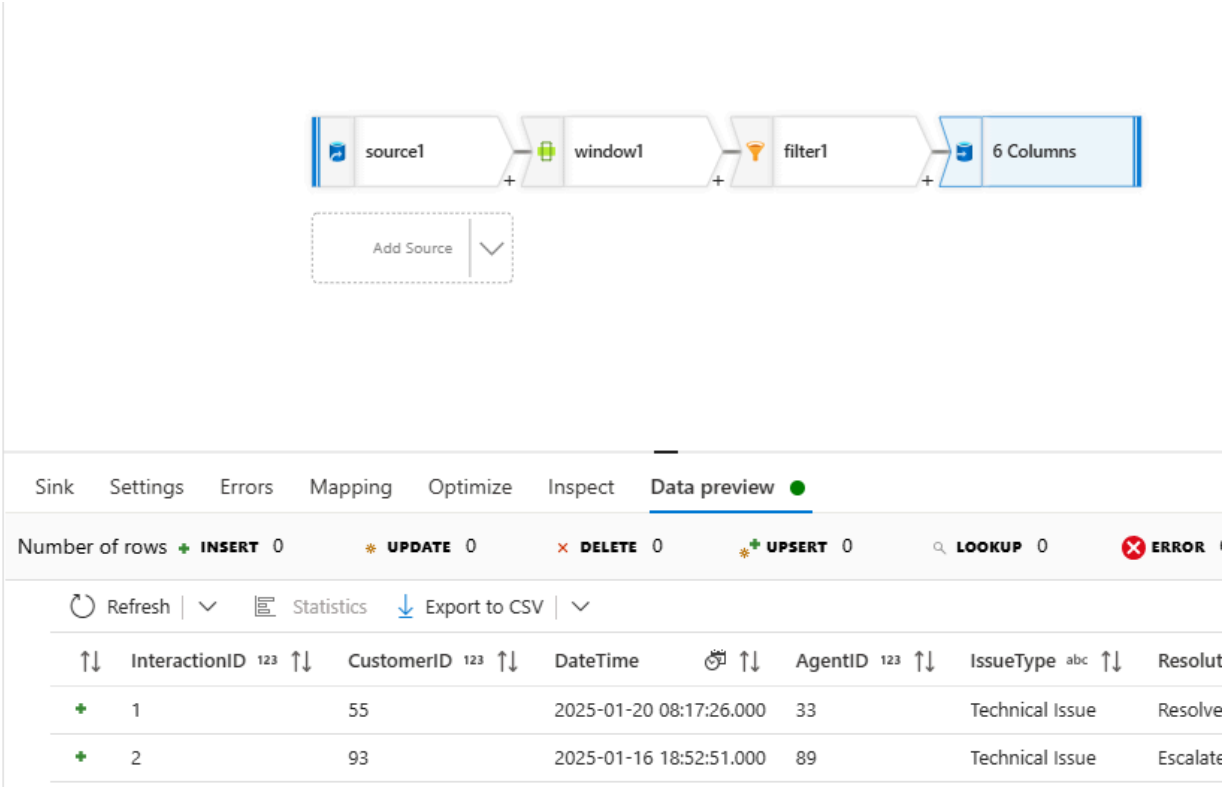


I put all the independent files in a single dataflow and those having Foreign Key are separate.

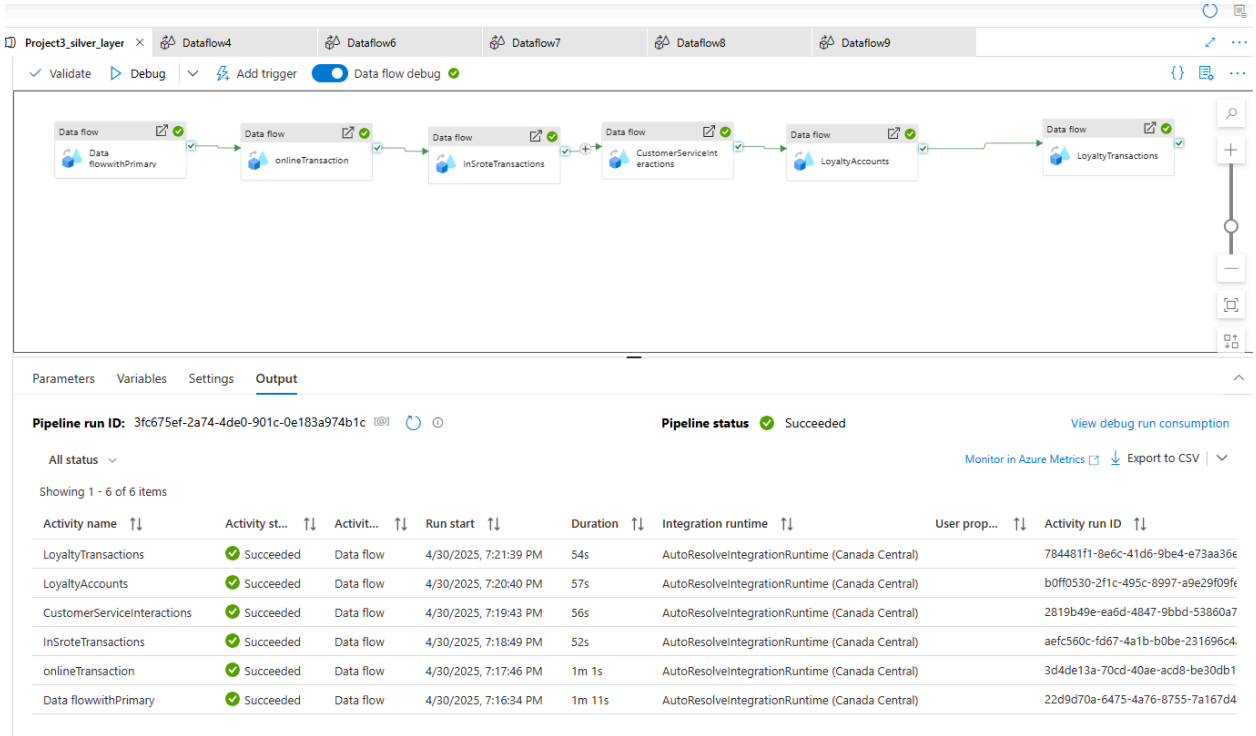

Customers
window1
filter1
select1
sink1
Products
window2
filter2
select2
sink2
Agents
window3
filter3
select3
sink3
Stores
window4
filter4
select4
sink4


ata flow
onlineTransaction
Data flow
InSroteTransactions
Data flow
CustomerServiceInteractions

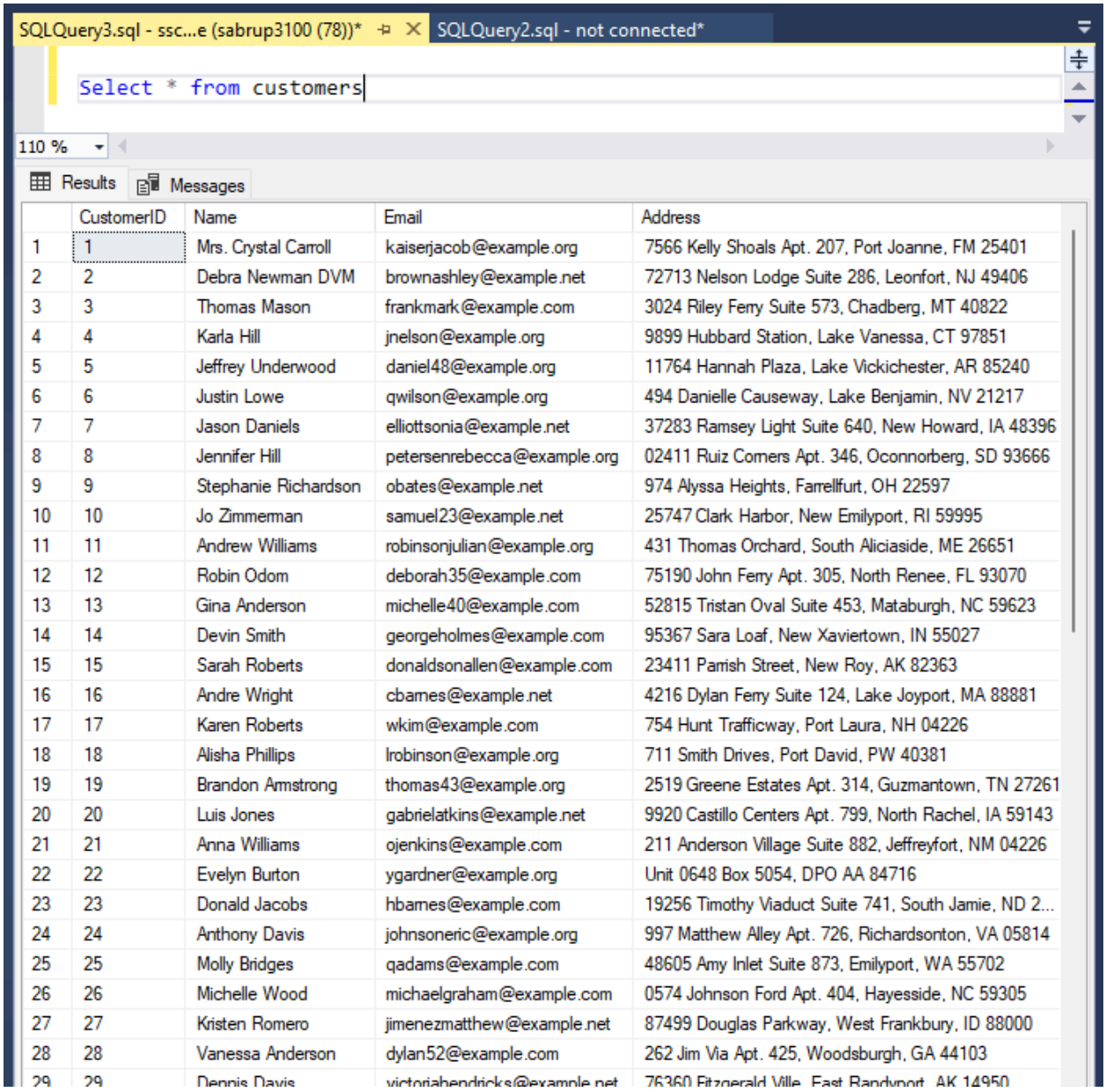
6. **And used same transformations as showing above:**



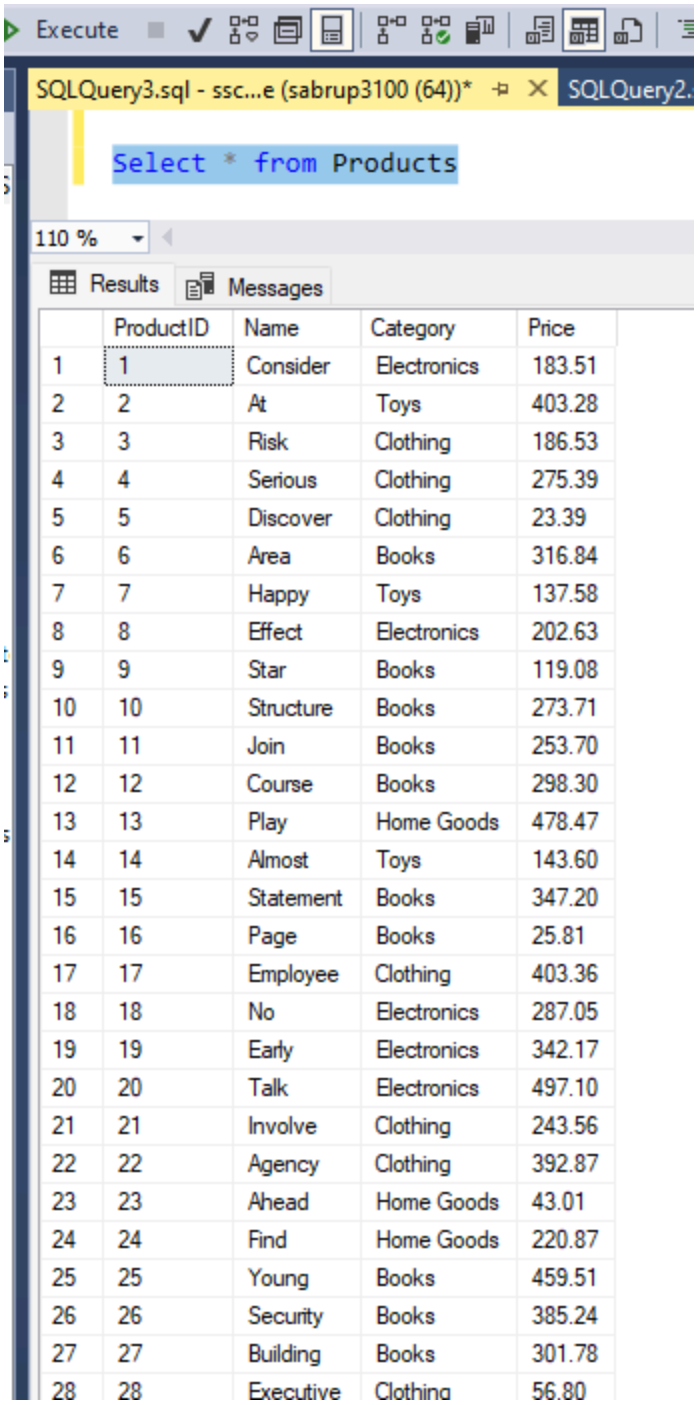
Here's the successfully running dataflows:



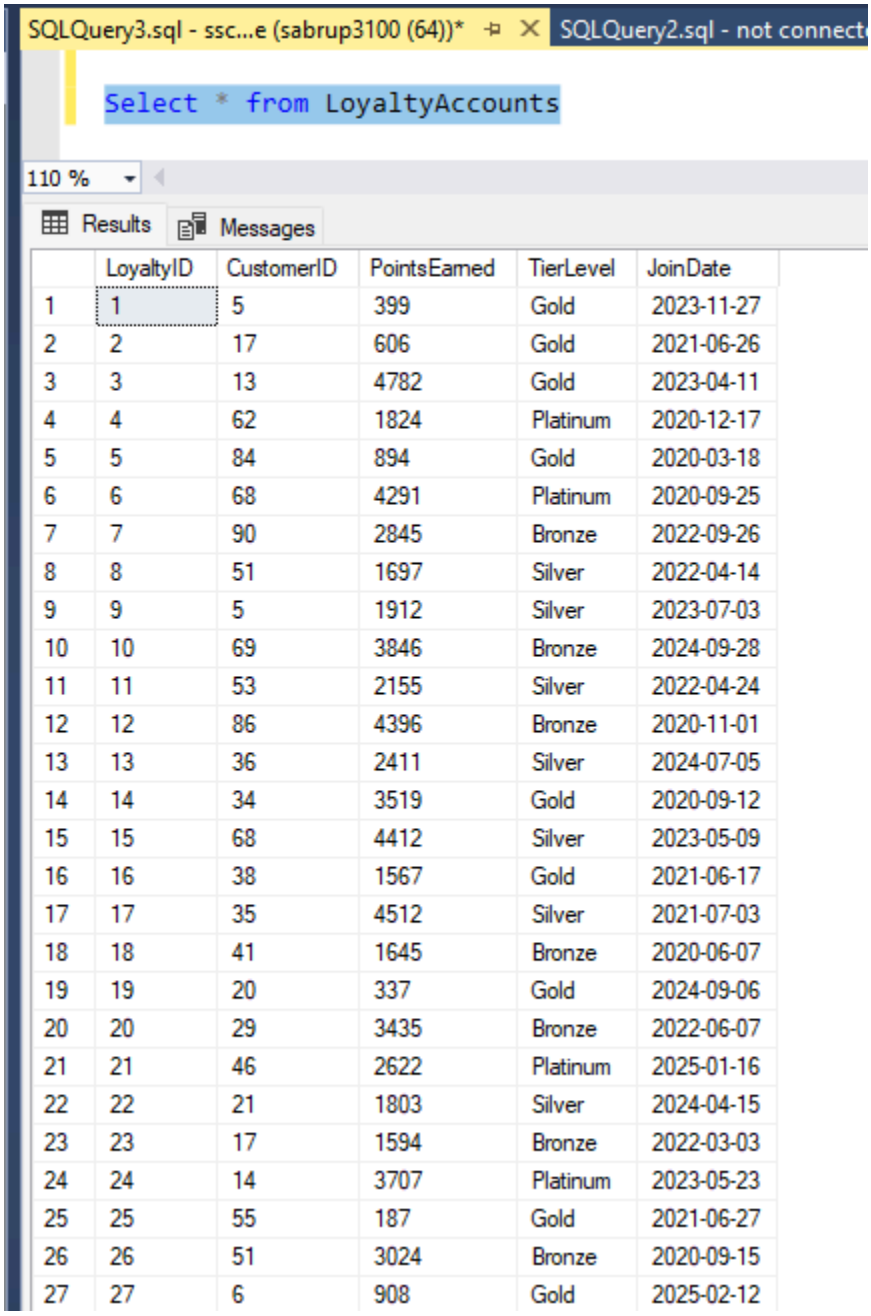
The data in the tables:



	CustomerID	Name	Email	Address
1	1	Mrs. Crystal Carroll	kaiserjacob@example.org	7566 Kelly Shoals Apt. 207, Port Joanne, FM 25401
2	2	Debra Newman DVM	brownashley@example.net	72713 Nelson Lodge Suite 286, Leonfort, NJ 49406
3	3	Thomas Mason	frankmark@example.com	3024 Riley Ferry Suite 573, Chadberg, MT 40822
4	4	Karla Hill	jnelson@example.org	9899 Hubbard Station, Lake Vanessa, CT 97851
5	5	Jeffrey Underwood	daniel48@example.org	11764 Hannah Plaza, Lake Vickichester, AR 85240
6	6	Justin Lowe	qwilson@example.org	494 Danielle Causeway, Lake Benjamin, NV 21217
7	7	Jason Daniels	elliottsonia@example.net	37283 Ramsey Light Suite 640, New Howard, IA 48396
8	8	Jennifer Hill	petersenrebecca@example.org	02411 Ruiz Corners Apt. 346, Oconnorberg, SD 93666
9	9	Stephanie Richardson	obates@example.net	974 Alyssa Heights, Farrellfurt, OH 22597
10	10	Jo Zimmerman	samuel23@example.net	25747 Clark Harbor, New Emilyport, RI 59995
11	11	Andrew Williams	robinsonjulian@example.org	431 Thomas Orchard, South Aliciaside, ME 26651
12	12	Robin Odom	deborah35@example.com	75190 John Ferry Apt. 305, North Renee, FL 93070
13	13	Gina Anderson	michelle40@example.com	52815 Tristan Oval Suite 453, Mataburgh, NC 59623
14	14	Devin Smith	georgeholmes@example.com	95367 Sara Loaf, New Xaviertown, IN 55027
15	15	Sarah Roberts	donaldsonallen@example.com	23411 Parrish Street, New Roy, AK 82363
16	16	Andre Wright	cbarnes@example.net	4216 Dylan Ferry Suite 124, Lake Joyport, MA 88881
17	17	Karen Roberts	wkim@example.com	754 Hunt Trafficway, Port Laura, NH 04226
18	18	Alisha Phillips	lrobinson@example.org	711 Smith Drives, Port David, PW 40381
19	19	Brandon Armstrong	thomas43@example.org	2519 Greene Estates Apt. 314, Guzmantown, TN 27261
20	20	Luis Jones	gabrielatkins@example.net	9920 Castillo Centers Apt. 799, North Rachel, IA 59143
21	21	Anna Williams	ojenkins@example.com	211 Anderson Village Suite 882, Jeffreyfort, NM 04226
22	22	Evelyn Burton	ygardner@example.org	Unit 0648 Box 5054, DPO AA 84716
23	23	Donald Jacobs	hbarnes@example.com	19256 Timothy Viaduct Suite 741, South Jamie, ND 2...
24	24	Anthony Davis	johnsoneric@example.org	997 Matthew Alley Apt. 726, Richardsonton, VA 05814
25	25	Molly Bridges	qadams@example.com	48605 Amy Inlet Suite 873, Emilyport, WA 55702
26	26	Michelle Wood	michaelgraham@example.com	0574 Johnson Ford Apt. 404, Hayesside, NC 59305
27	27	Kristen Romero	jimenezmatthew@example.net	87499 Douglas Parkway, West Frankbury, ID 88000
28	28	Vanessa Anderson	dylan52@example.com	262 Jim Via Apt. 425, Woodsburgh, GA 44103
29	29	Dennis Davis	victoriahendricks@example.net	76360 Fitzgerald Ville, East Randyport, AK 14950



	ProductID	Name	Category	Price
1	1	Consider	Electronics	183.51
2	2	At	Toys	403.28
3	3	Risk	Clothing	186.53
4	4	Serious	Clothing	275.39
5	5	Discover	Clothing	23.39
6	6	Area	Books	316.84
7	7	Happy	Toys	137.58
8	8	Effect	Electronics	202.63
9	9	Star	Books	119.08
10	10	Structure	Books	273.71
11	11	Join	Books	253.70
12	12	Course	Books	298.30
13	13	Play	Home Goods	478.47
14	14	Almost	Toys	143.60
15	15	Statement	Books	347.20
16	16	Page	Books	25.81
17	17	Employee	Clothing	403.36
18	18	No	Electronics	287.05
19	19	Early	Electronics	342.17
20	20	Talk	Electronics	497.10
21	21	Involve	Clothing	243.56
22	22	Agency	Clothing	392.87
23	23	Ahead	Home Goods	43.01
24	24	Find	Home Goods	220.87
25	25	Young	Books	459.51
26	26	Security	Books	385.24
27	27	Building	Books	301.78
28	28	Executive	Clothing	56.80



	LoyaltyID	CustomerID	PointsEarned	TierLevel	JoinDate
1	1	5	399	Gold	2023-11-27
2	2	17	606	Gold	2021-06-26
3	3	13	4782	Gold	2023-04-11
4	4	62	1824	Platinum	2020-12-17
5	5	84	894	Gold	2020-03-18
6	6	68	4291	Platinum	2020-09-25
7	7	90	2845	Bronze	2022-09-26
8	8	51	1697	Silver	2022-04-14
9	9	5	1912	Silver	2023-07-03
10	10	69	3846	Bronze	2024-09-28
11	11	53	2155	Silver	2022-04-24
12	12	86	4396	Bronze	2020-11-01
13	13	36	2411	Silver	2024-07-05
14	14	34	3519	Gold	2020-09-12
15	15	68	4412	Silver	2023-05-09
16	16	38	1567	Gold	2021-06-17
17	17	35	4512	Silver	2021-07-03
18	18	41	1645	Bronze	2020-06-07
19	19	20	337	Gold	2024-09-06
20	20	29	3435	Bronze	2022-06-07
21	21	46	2622	Platinum	2025-01-16
22	22	21	1803	Silver	2024-04-15
23	23	17	1594	Bronze	2022-03-03
24	24	14	3707	Platinum	2023-05-23
25	25	55	187	Gold	2021-06-27
26	26	51	3024	Bronze	2020-09-15
27	27	6	908	Gold	2025-02-12

Step 5: Load Analytics Data

View 1:

```
CREATE VIEW View_AOV AS
SELECT
    p.Name AS Product_Name,
    p.Category,
    s.Location,
    COUNT(ot.OrderID) AS Total_Orders,
    SUM(ot.Amount) AS TotalAmount,
    SUM(ot.Amount) / COUNT(ot.OrderID) AS AverageOrderValue
FROM
    OnlineTransactions ot
JOIN Products p ON ot.ProductID = p.ProductID
JOIN InStoreTransactions ist ON ot.CustomerID = ist.CustomerID
JOIN Stores s ON s.StoreID = ist.StoreID
GROUP BY s.Location, p.Category, p.Name;

select * from View_AOV
```

110 %

Results | Messages

	Product_Name	Category	Location	Total_Orders	TotalAmount	AverageOrderValue
1	Alone	Books	Alexandraside	1	187.64	187.640000
2	Drug	Clothing	Alexandraside	1	199.99	199.990000
3	Subject	Electronics	Alexandraside	1	115.71	115.710000
4	Ahead	Home Goods	Ashleyfort	1	149.00	149.000000
5	Hand	Electronics	Ayersville	1	141.15	141.150000
6	Other	Clothing	Bakerberg	1	103.67	103.670000
7	Almost	Toys	Bakerberg	1	178.32	178.320000
8	Play	Home Goods	Bishopview	1	147.49	147.490000
9	Tend	Books	Chapmanhaven	1	17.41	17.410000
10	Alone	Books	Dariusberg	1	45.51	45.510000
11	Sign	Books	East Justin	1	25.96	25.960000
12	Effect	Electronics	East Justin	1	166.46	166.460000
13	Happy	Toys	East Justin	1	173.83	173.830000
14	Early	Electronics	East Laura	1	175.37	175.370000
15	Describe	Clothing	Ellisstad	1	44.52	44.520000
16	Truth	Books	Gabrielafurt	1	199.26	199.260000
17	Wish	Books	Gabrielafurt	1	136.61	136.610000
18	Almost	Home Goods	Haroldmouth	1	110.29	110.290000
19	Much	Home Goods	Haroldmouth	1	97.94	97.940000

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."**

SQLQuery4.sql - ssc...e (sabrup3100 (60))* | SQLQuery3.sql - ssc...e (sabrup3100 (64))*

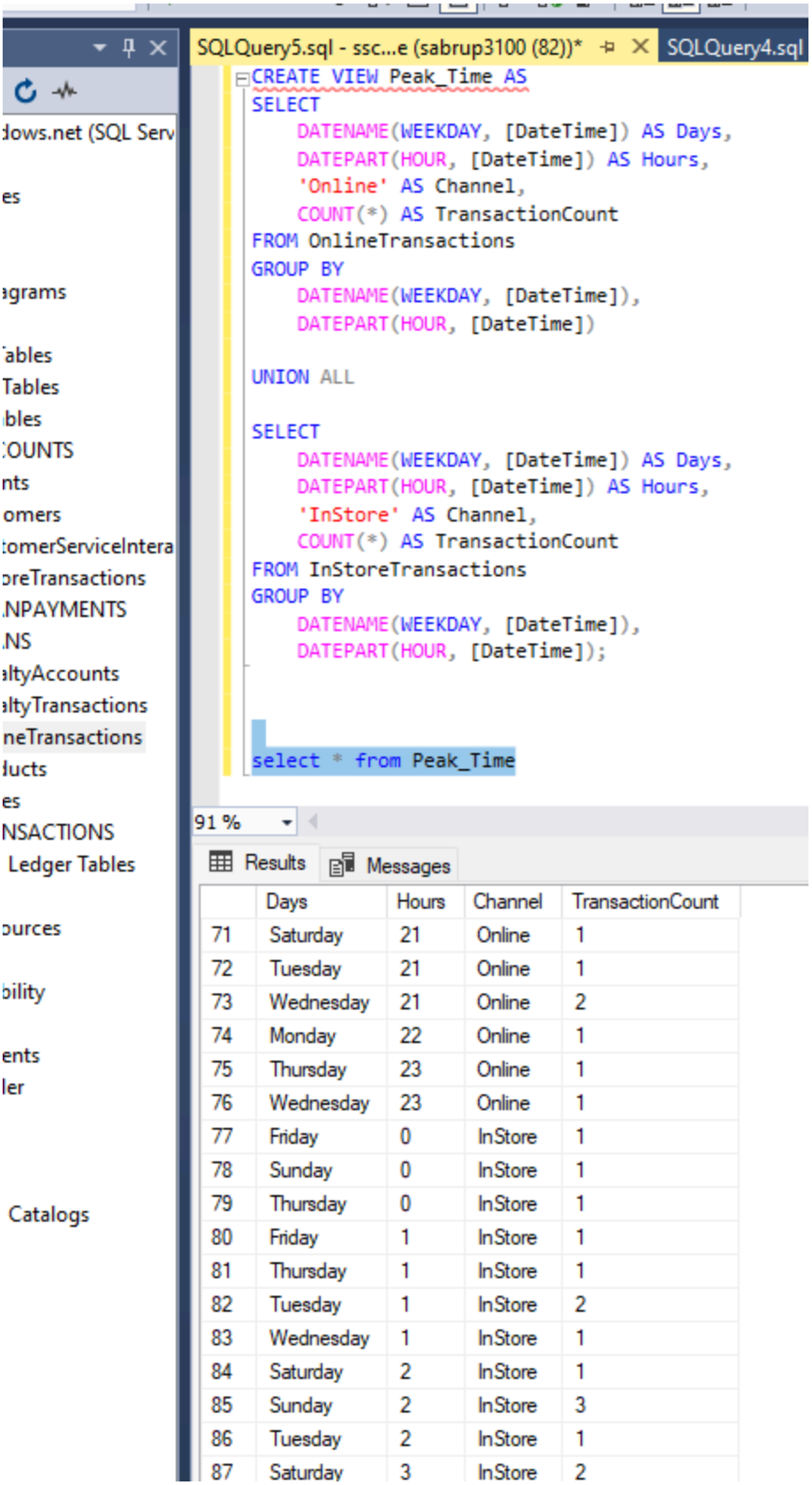
```sql
CREATE VIEW View_CustomerSegmentation AS
WITH CustomerSpendings AS (
    SELECT
        c.CustomerID,
        c.Name,
        SUM(ot.Amount) AS TotalSpendings,
        COUNT(ot.OrderID) AS Frequency,
        l.TierLevel
    FROM
        customers c
        LEFT JOIN OnlineTransactions ot ON c.CustomerID = ot.CustomerID
        LEFT JOIN LoyaltyAccounts l ON l.CustomerID = c.CustomerID
    GROUP BY
        c.CustomerID, c.Name, l.TierLevel
),
RankedCustomers AS (
    SELECT *,
           PERCENT_RANK() OVER (ORDER BY TotalSpendings DESC) AS SpendingRank
    FROM CustomerSpendings
)

SELECT
    CustomerID,
    Name,
    TotalSpendings,
    Frequency,
    TierLevel,
    CASE
        WHEN SpendingRank <= 0.1 THEN 'High-Value Customer'
        WHEN Frequency = 1 THEN 'One-Time Buyer'
        WHEN TierLevel IN ('Gold', 'Platinum') THEN 'Loyalty Champion'
        ELSE 'Regular'
    END AS CustomerSegmentation
FROM RankedCustomers;
```

91 %

Results | Messages

	CustomerID	Name	TotalSpendings	Frequency	TierLevel	CustomerSegmentation
1	9	Stephanie Richardson	1058.88	9	Platinum	High-Value Customer
2	13	Gina Anderson	663.24	4	Gold	High-Value Customer
3	63	Tony Jenkins DDS	474.52	4	Gold	High-Value Customer
4	51	Tyler Knight	460.89	3	Bronze	High-Value Customer
5	99	Kristina Mcmillan	353.54	2	NULL	High-Value Customer
6	90	Katherine Mccarthy	350.74	2	Bronze	High-Value Customer
7	100	Corey Romero	349.60	2	Silver	High-Value Customer
8	18	Alisha Phillips	335.87	2	Silver	High-Value Customer
9	44	Elizabeth Taylor	333.78	2	Platinum	High-Value Customer
10	44	Elizabeth Taylor	333.78	2	Gold	High-Value Customer
11	91	Rachel Lewis	332.92	2	Silver	High-Value Customer
12	12	Robin Odom	315.70	2	Bronze	High-Value Customer

- **View 3 - for Analyze `DateTime` to find peak days and times in-store vs. online.**



SQLQuery5.sql - ssc...e (sabrup3100 (82))* | SQLQuery4.sql

```
CREATE VIEW Peak_Time AS
SELECT
    DATENAME(WEEKDAY, [DateTime]) AS Days,
    DATEPART(HOUR, [DateTime]) AS Hours,
    'Online' AS Channel,
    COUNT(*) AS TransactionCount
FROM OnlineTransactions
GROUP BY
    DATENAME(WEEKDAY, [DateTime]),
    DATEPART(HOUR, [DateTime])

UNION ALL

SELECT
    DATENAME(WEEKDAY, [DateTime]) AS Days,
    DATEPART(HOUR, [DateTime]) AS Hours,
    'InStore' AS Channel,
    COUNT(*) AS TransactionCount
FROM InStoreTransactions
GROUP BY
    DATENAME(WEEKDAY, [DateTime]),
    DATEPART(HOUR, [DateTime]);


select * from Peak_Time
```

91 %

Results | Messages

	Days	Hours	Channel	TransactionCount
71	Saturday	21	Online	1
72	Tuesday	21	Online	1
73	Wednesday	21	Online	2
74	Monday	22	Online	1
75	Thursday	23	Online	1
76	Wednesday	23	Online	1
77	Friday	0	InStore	1
78	Sunday	0	InStore	1
79	Thursday	0	InStore	1
80	Friday	1	InStore	1
81	Thursday	1	InStore	1
82	Tuesday	1	InStore	2
83	Wednesday	1	InStore	1
84	Saturday	2	InStore	1
85	Sunday	2	InStore	3
86	Tuesday	2	InStore	1
87	Saturday	3	InStore	2

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

```
CREATE VIEW VIEW4 AS
SELECT
     a.AgentID ,
     a.Name AS AgentName ,
     COUNT(csi.InteractionID) AS InteractionCount,
     SUM(CASE WHEN csi.ResolutionStatus = 'Resolved' THEN 1 ELSE 0 END) AS ResolvedCount,
     CASE
        WHEN COUNT(csi.InteractionID) = 0 THEN 0
        ELSE CAST(
            (SUM(CASE WHEN csi.ResolutionStatus = 'Ressolved' THEN 1 ELSE 0 END) *100.0)/
            COUNT(csi.InteractionID)
            AS DECIMAL(5,2)
            )
        END AS ResolutionSuccessRate
FROM Agents a
LEFT JOIN
CustomerServiceInteractions csi ON a.AgentID = csi.AgentID
GROUP BY a.AgentID, a.NAme

SELECT * FROM VIEW4
```

100 %

Results | Messages

	AgentID	AgentName	InteractionCount	ResolvedCount	ResolutionSuccessRate
1	1	Jonathan Williams	1	0	0.00
2	2	Terry Edwards	1	0	0.00
3	3	Garrett Knapp	0	0	0.00
4	4	Daryl Benjamin	2	0	0.00
5	5	Matthew Long	1	0	0.00
6	6	Patricia Rhodes	0	0	0.00
7	7	Elizabeth James	2	2	0.00
8	8	Teresa Bennett	1	0	0.00
9	9	Amber Ross	2	1	0.00
10	10	Tonya Jones	1	0	0.00
11	11	Curtis Mcbride	1	1	0.00
12	12	Justin Michael	0	0	0.00
13	13	Scott Flowers	1	0	0.00
14	14	Michael Jackson	0	0	0.00
15	15	Kristen Crawford	1	1	0.00
16	16	Jo Meyers	0	0	0.00
17	17	Brandon Jimenez	1	0	0.00
18	18	Melissa White	2	0	0.00
19	19	Eddie Pierce	1	0	0.00

Query executed successfu... | sscserver.database.windows.... | sabrup3100 (82) | sabrupdatabase | 00:00:00 | 100 row

Col 1 Ch 1 INS