**Data Pipeline for Customer Account Analysis Project**

**Objective:**

To design and implement a robust data pipeline that ingests customer account data from a backend source, performs data cleaning and transformation, and loads the final data using Slowly Changing Dimensions (SCD) logic. This supports downstream analytics and reporting via Power BI.

**Architecture Overview:**

* Azure Virtual Machine (on-premise simulation)
* Self-hosted Integration Runtime (IR)
* Azure Data Factory (ADF)
* Azure Data Lake Storage Gen2 (Bronze, Silver, Gold layers)
* Azure Key Vault (for secure credentials)
* Azure SQL Database
* Power BI for visualization

A diagram of a computer system

AI-generated content may be incorrect.

**Key Vault created with secrets:**

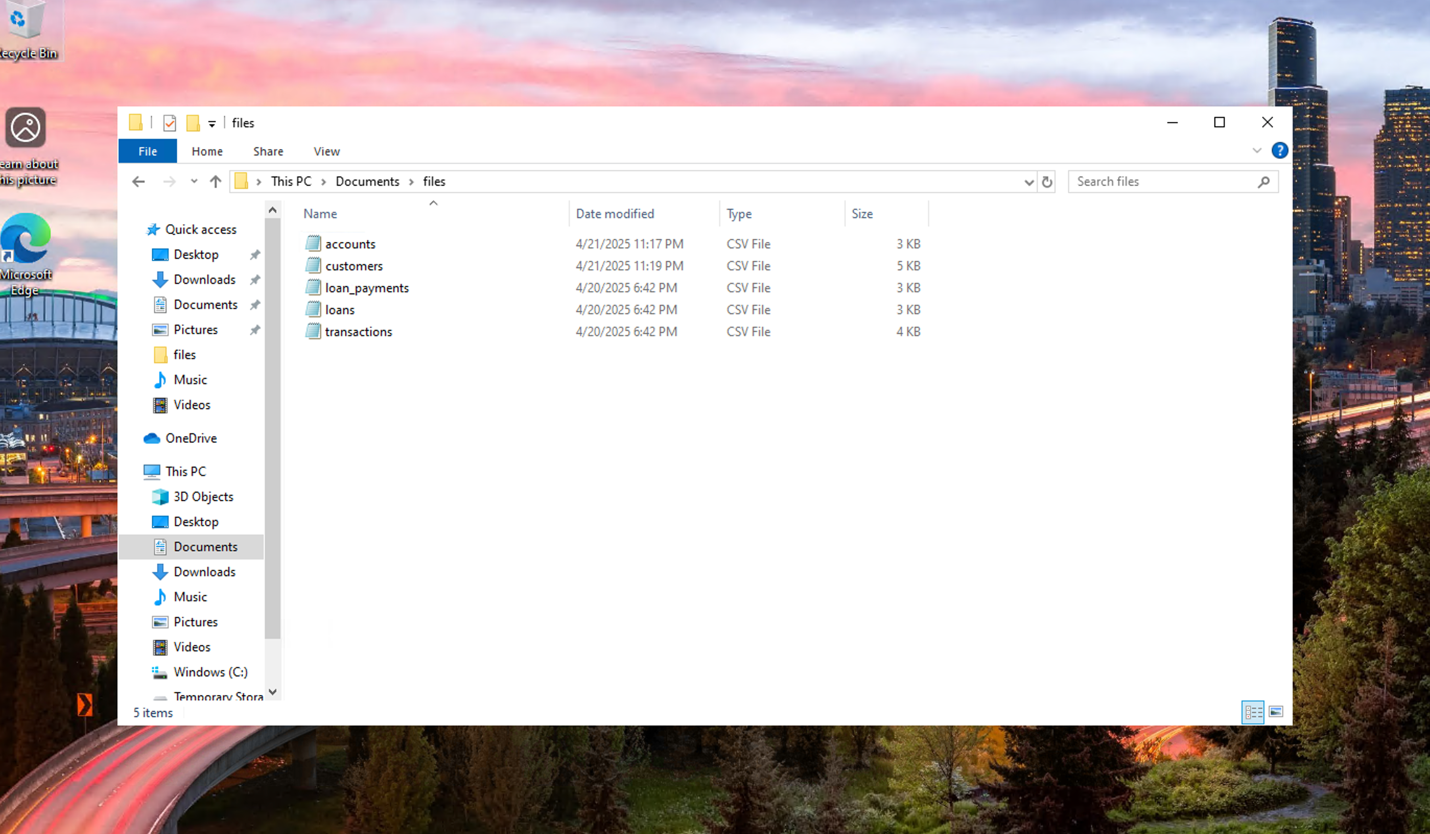
A screenshot of a computer

AI-generated content may be incorrect.

**Step 1:** **Data Ingestion (On-prem to Bronze Layer)**

**Provisioned Azure Virtual Machine:**

Simulated as on-premise environment.

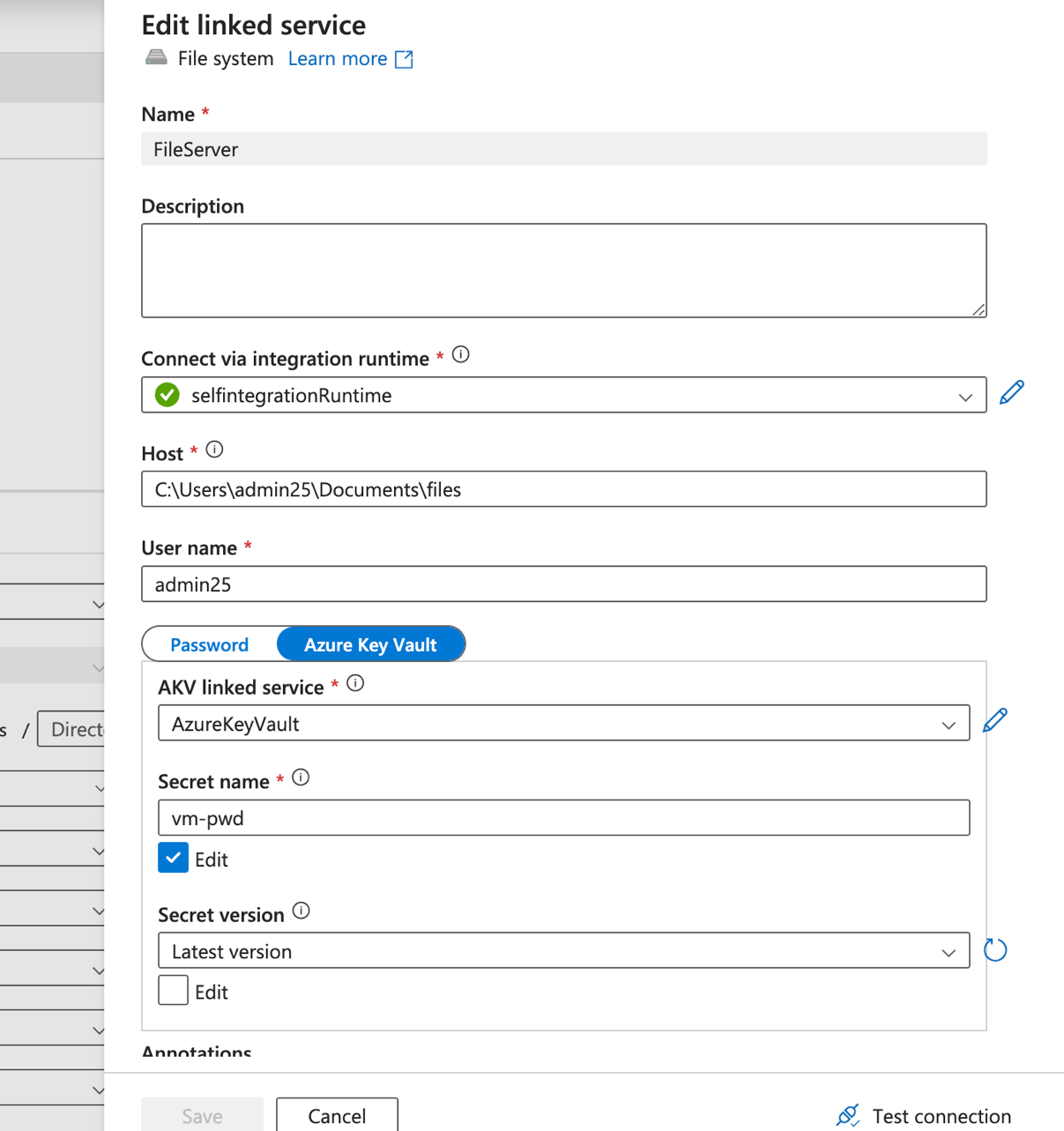


* Installed and configured Self- hosted Integration Runtime on-premise

A screenshot of a computer

AI-generated content may be incorrect.

* Azure Key Vault used to securely fetch VM password during configuration.

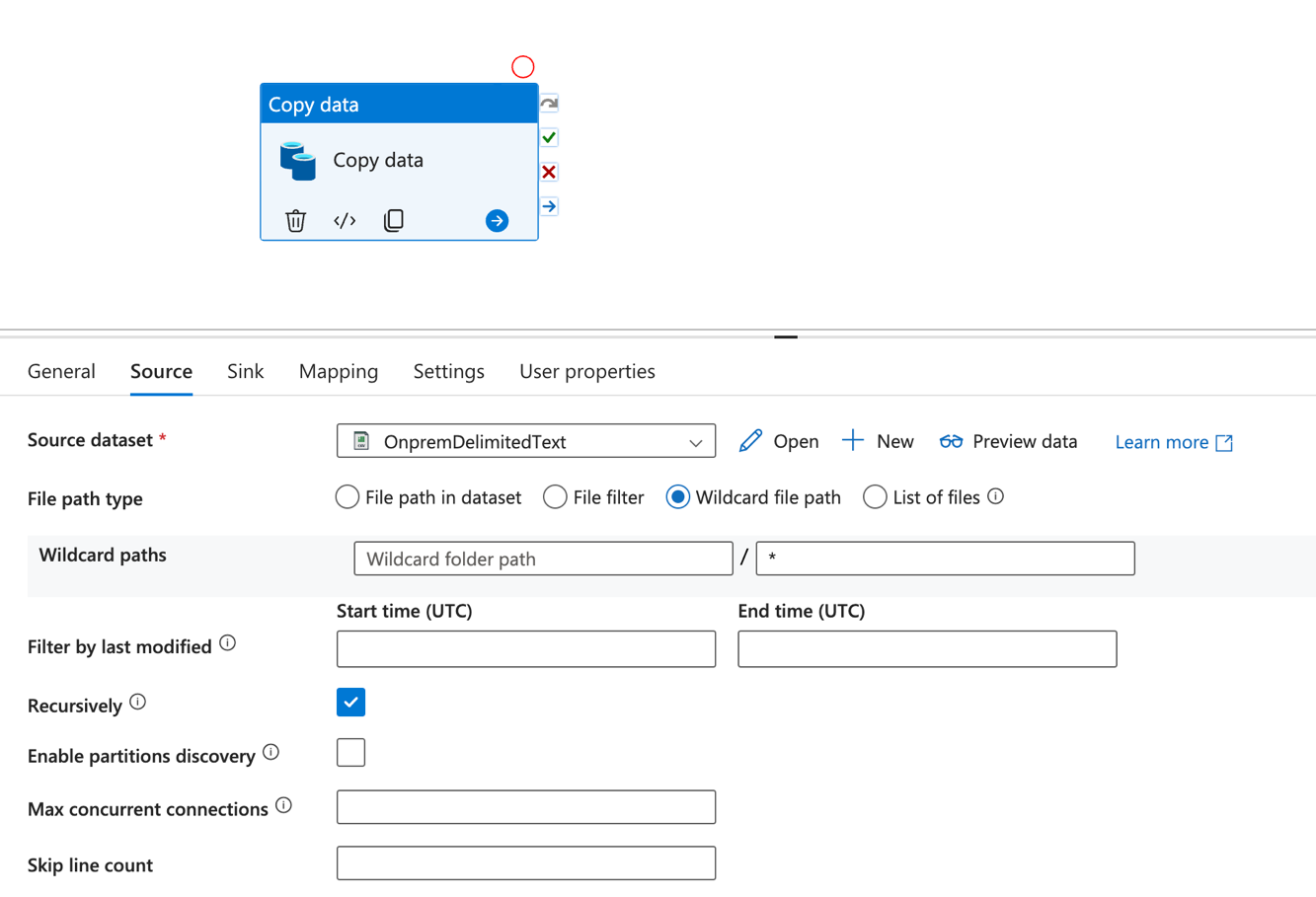


**ADF Copy Activity:**

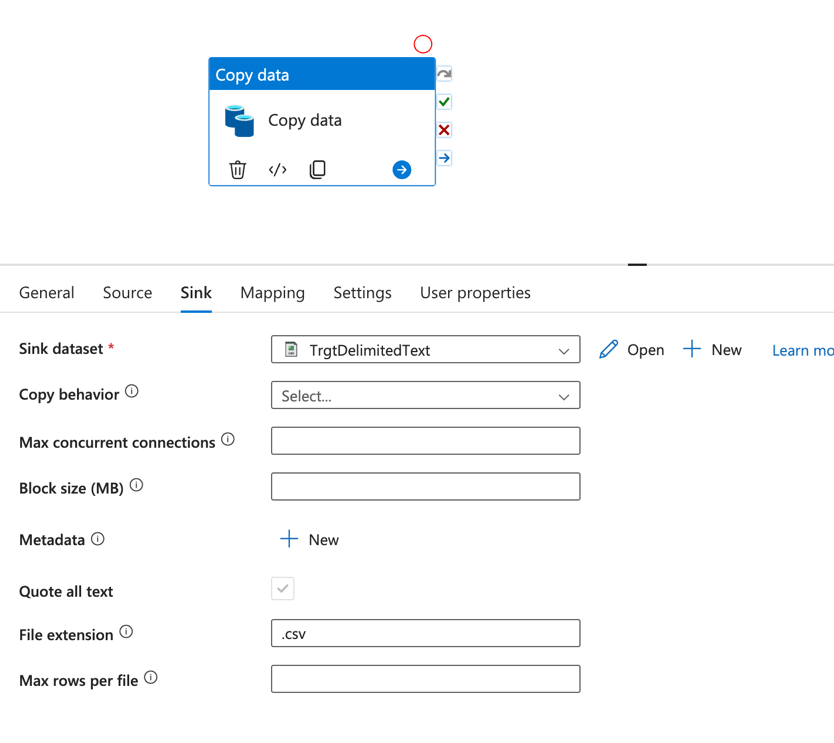
**Source**: File system on Azure VM.

A screenshot of a computer

AI-generated content may be incorrect.



**Sink:** ADLS Gen2 Bronze container.



A screenshot of a computer

AI-generated content may be incorrect.

**Files Ingested:**

accounts.csv

customers.csv

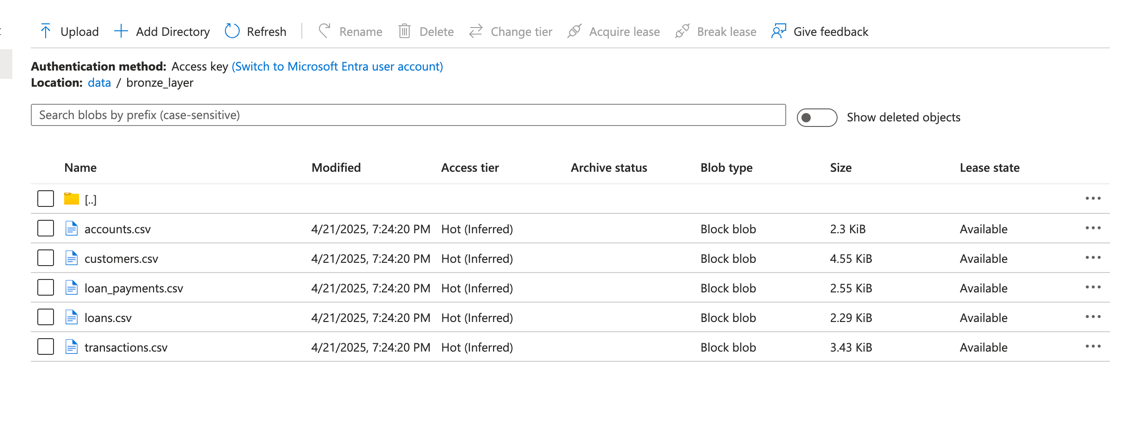
loan\_payments.csv

loans.csv

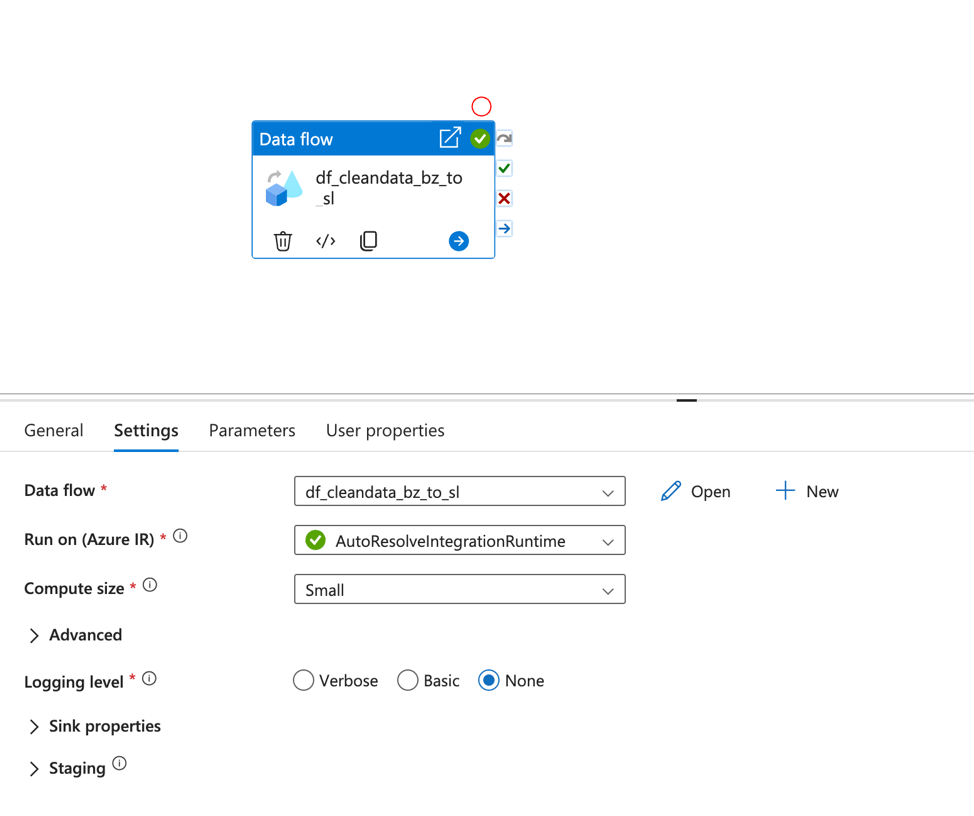
transactions.csv

**Outcome**:

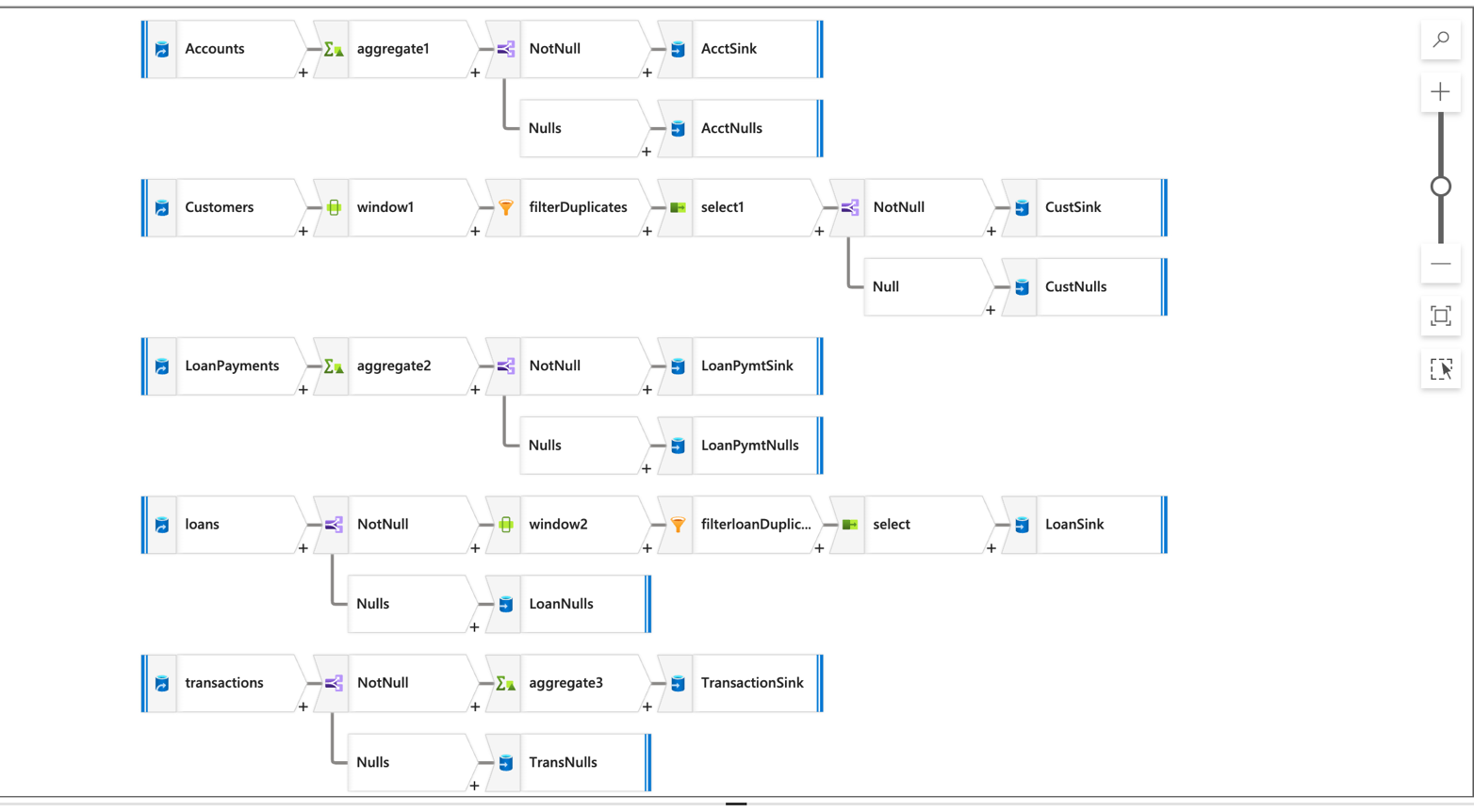
All raw files successfully copied into the Bronze layer of ADLS Gen2.



**Step 2: Data Cleaning & Transformation (Bronze to Silver Layer)**

Created New Pipeline with Data Flow in ADF:

**Read five datasets from the Bronze container.**

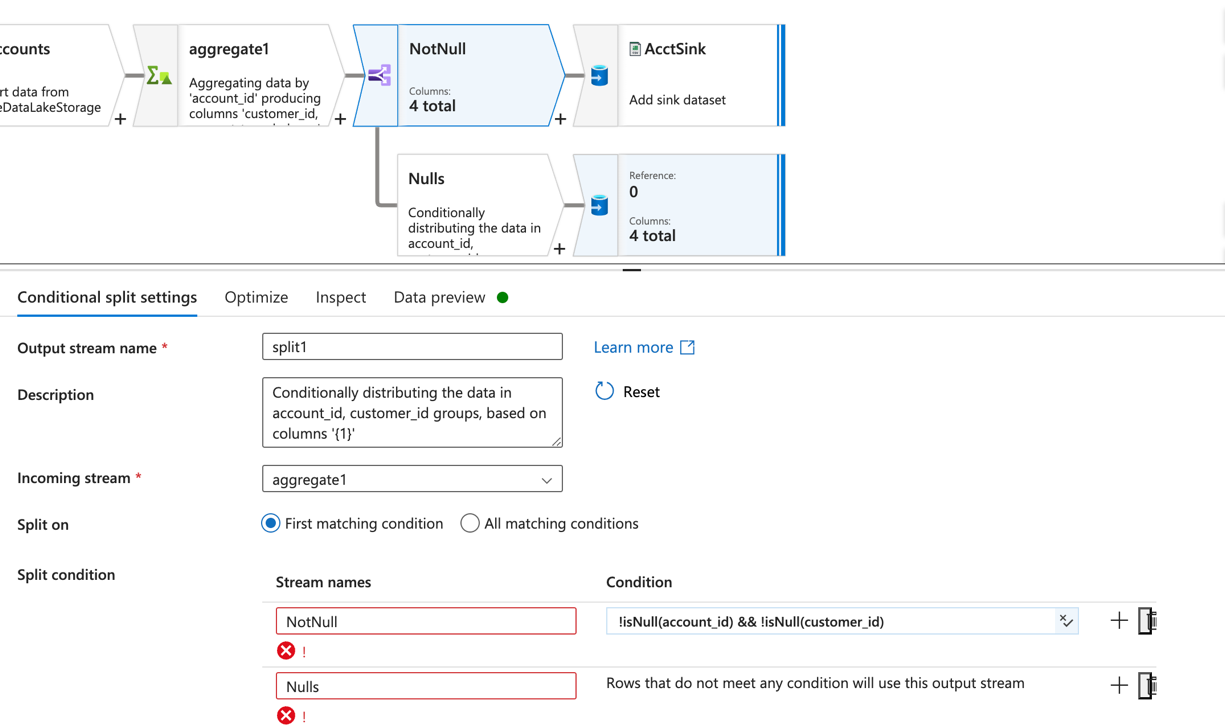


**Data Cleaning:**

**Conditional Split transformation used:**

**Expression**: !isNull(column) for necessary key column.

* This removed rows with null critical fields.



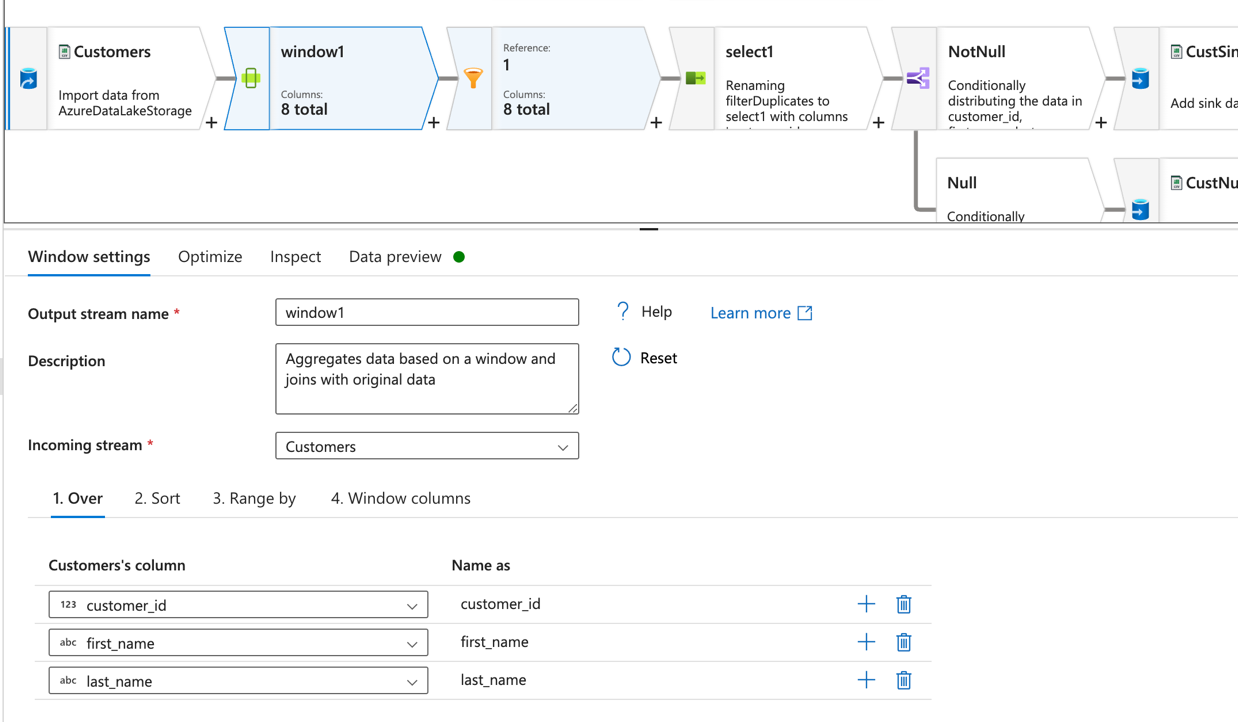
**Deduplication:**

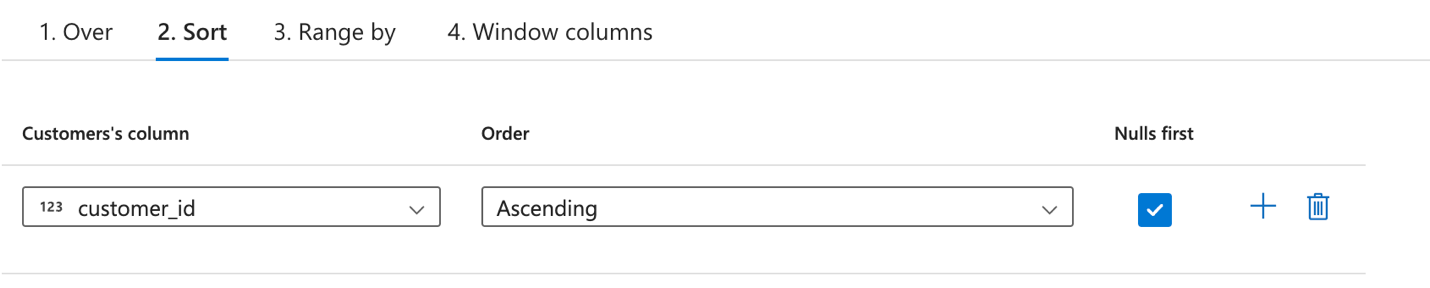
**Window Transformation used:**

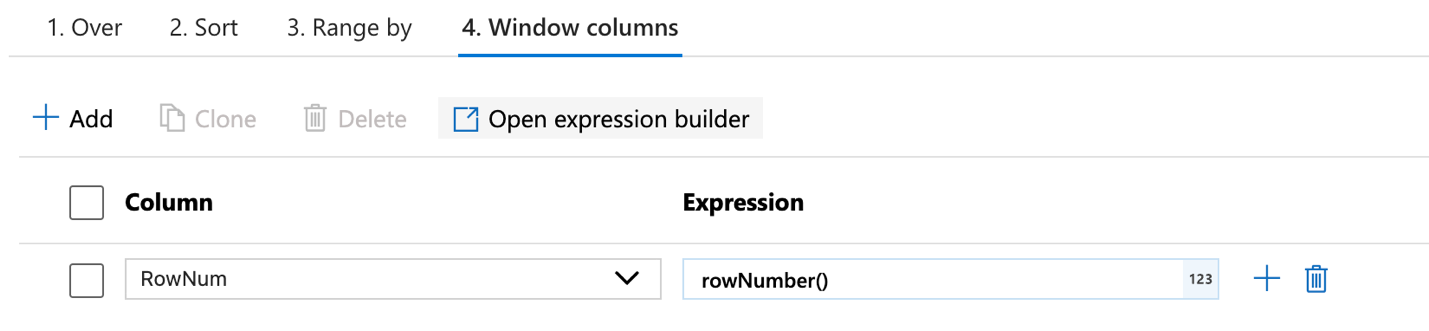
**Function:** RowNumber

Partitioned by key column (e.g., account\_id)

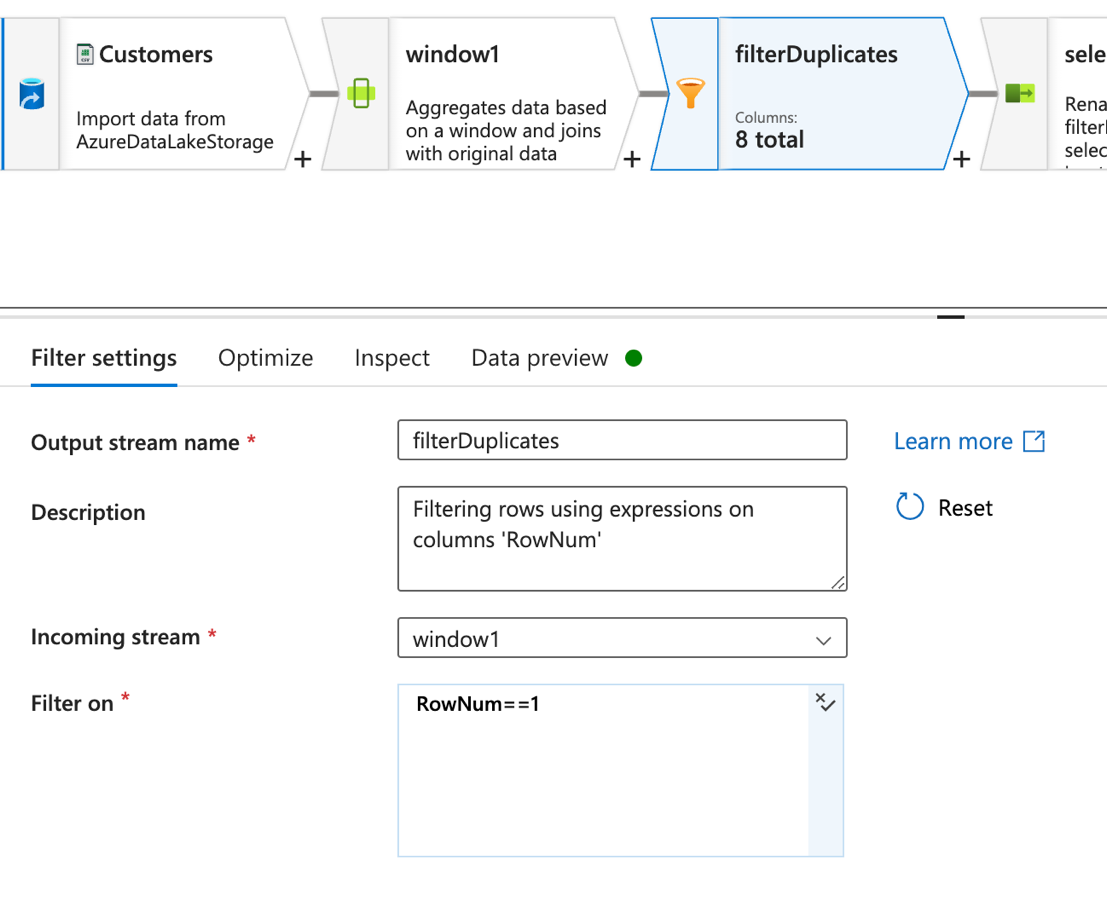
Kept only row\_number == 1 rows





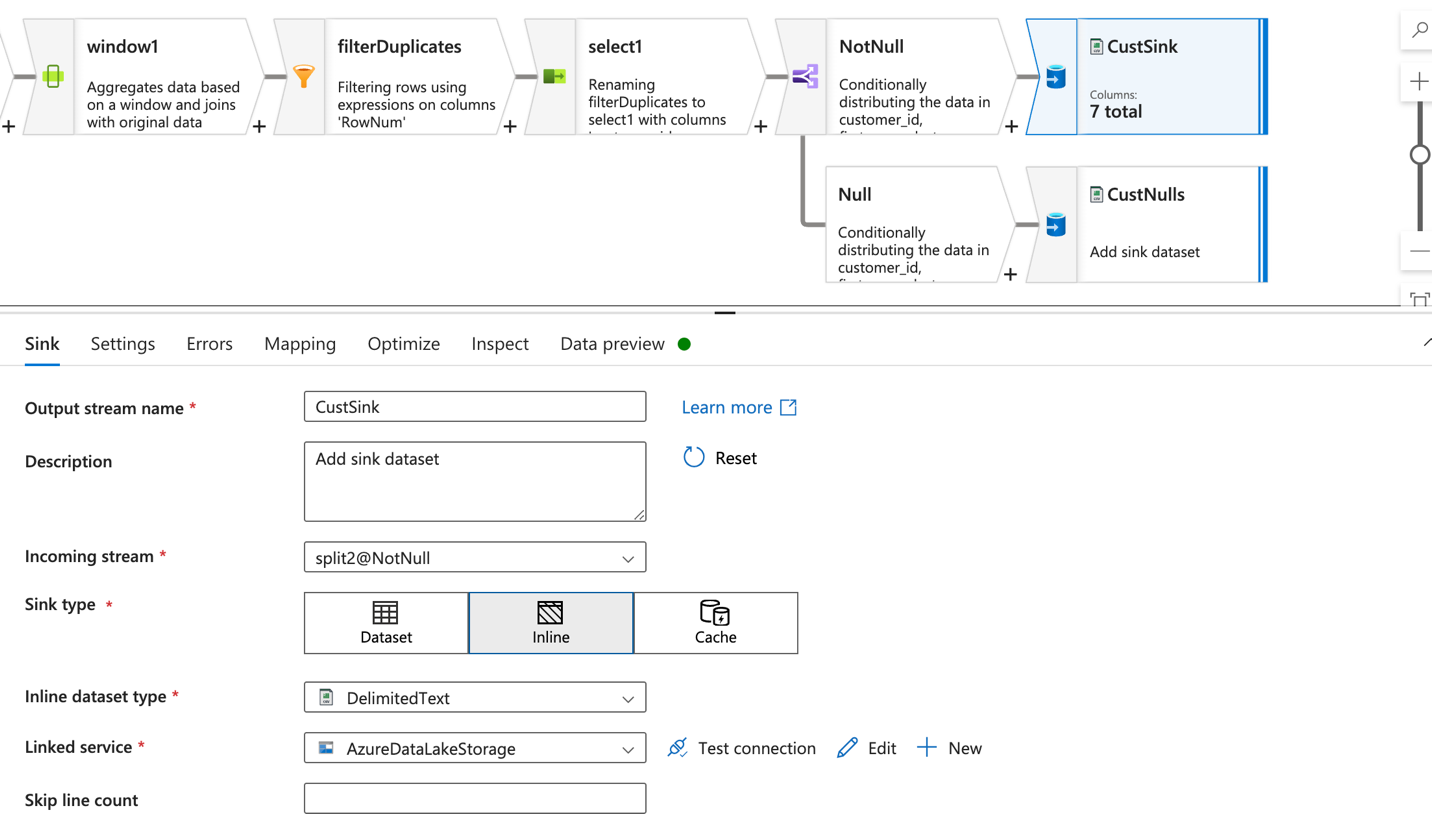


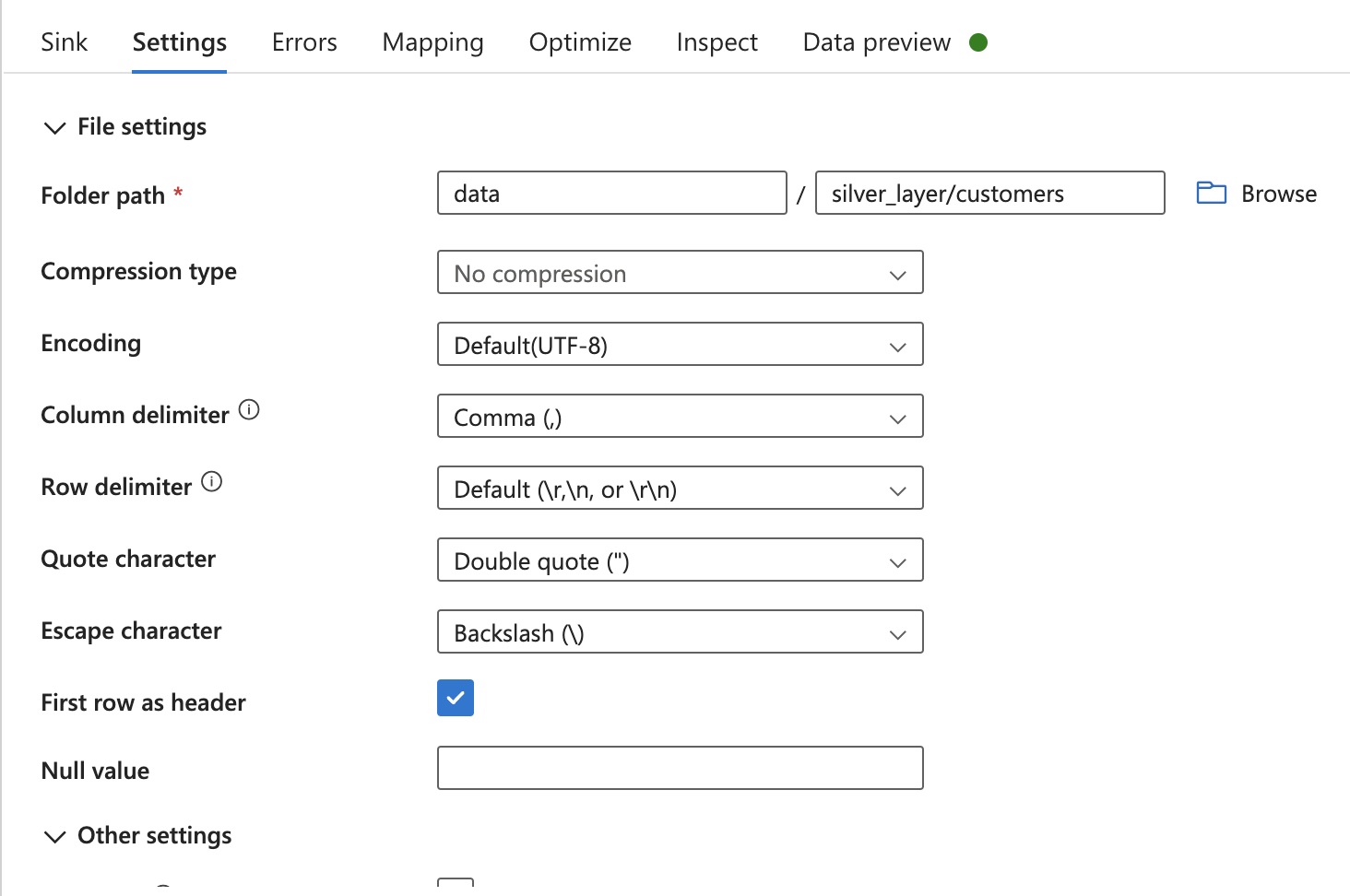
**Filter** transformation to filter the duplicate rows



**Sink:**

Cleaned data written to Silver container in ADLS Gen2.

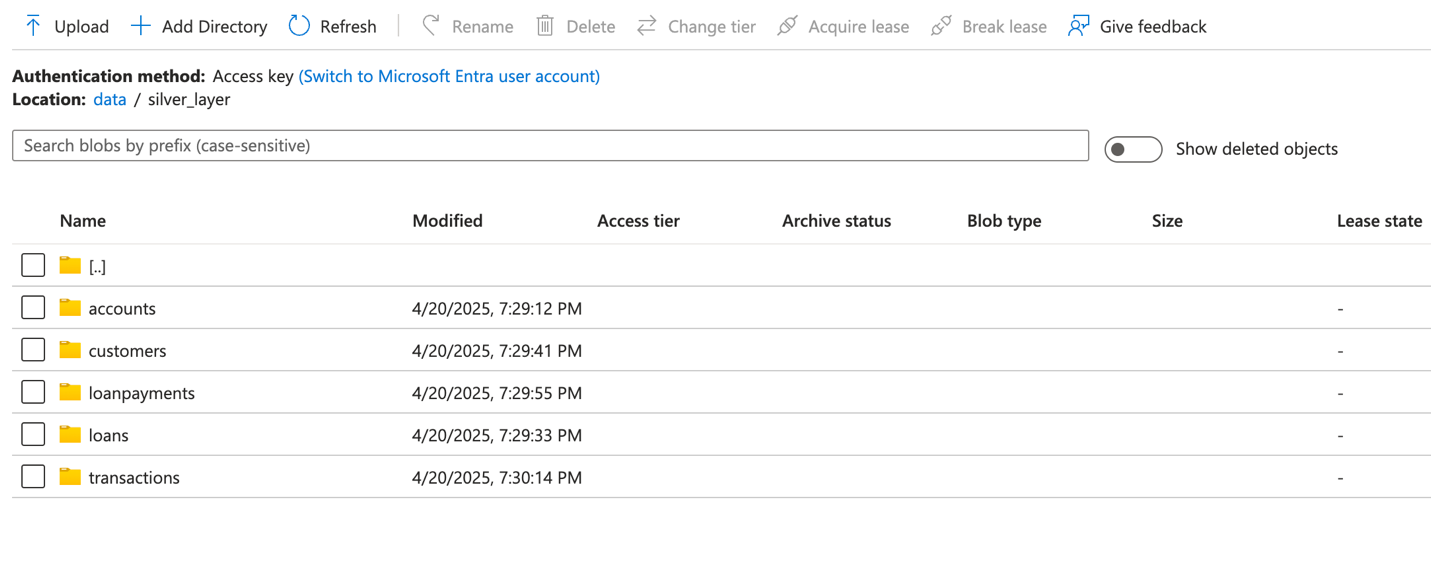


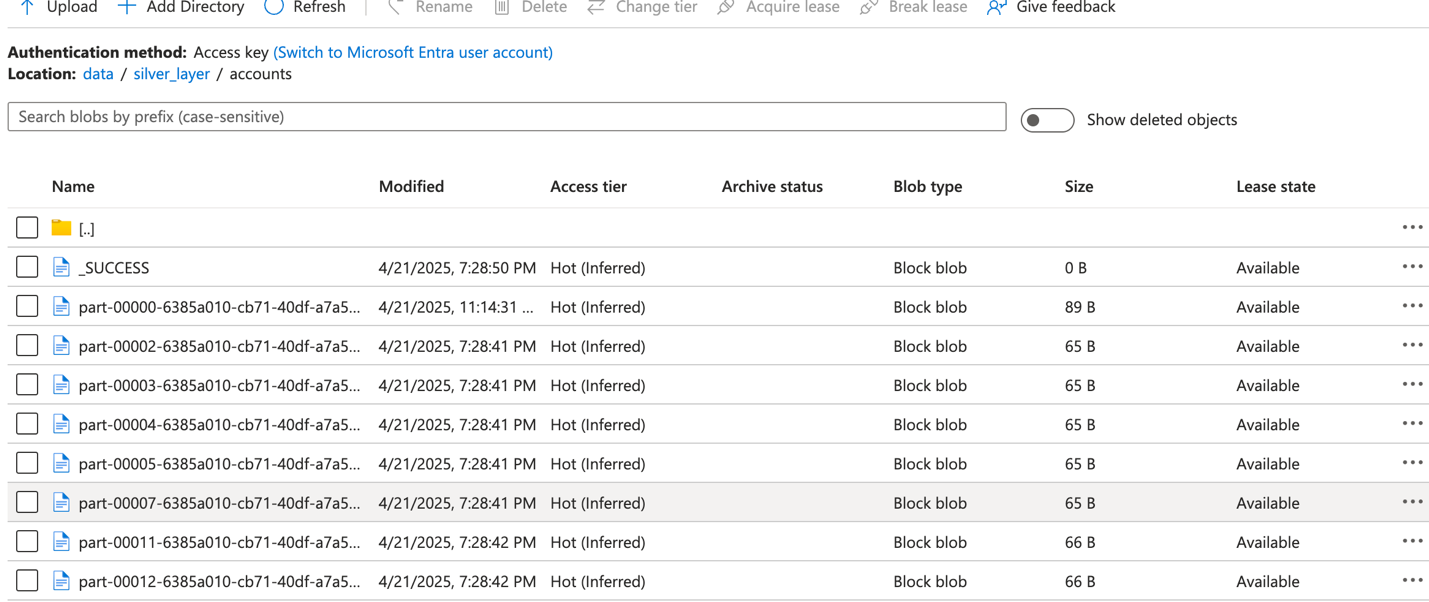


* Nulls Sink for storing all null values.



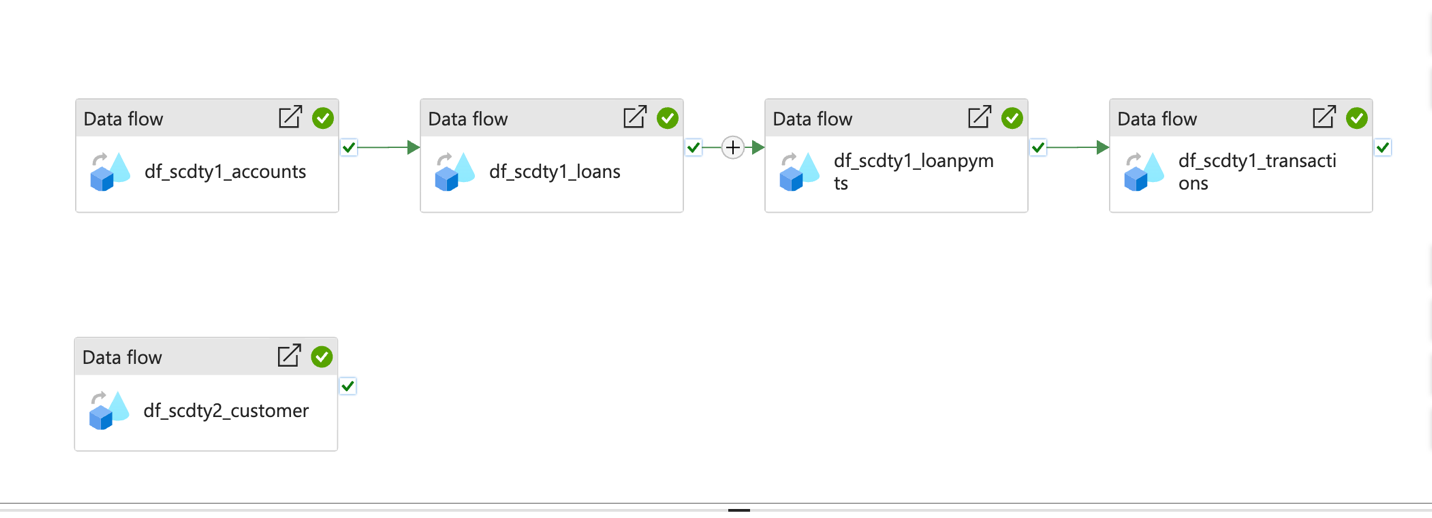
**SILVER LAYER –** Cleaned and transformed data.





**Step 3: SCD Logic & Gold Layer (Silver to Gold)**

Created Another Pipeline for SCD Logic using Data Flow:



**SCD Type 1 Implemented For:**

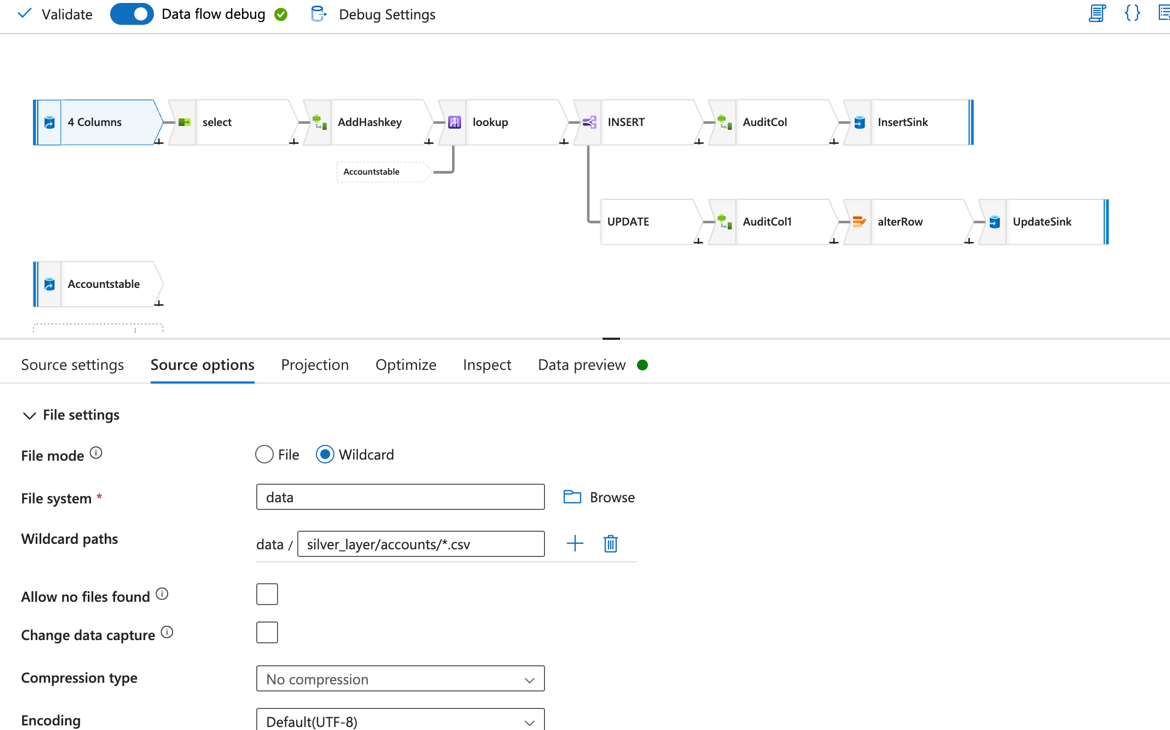
**accounts**

**loan\_payments**

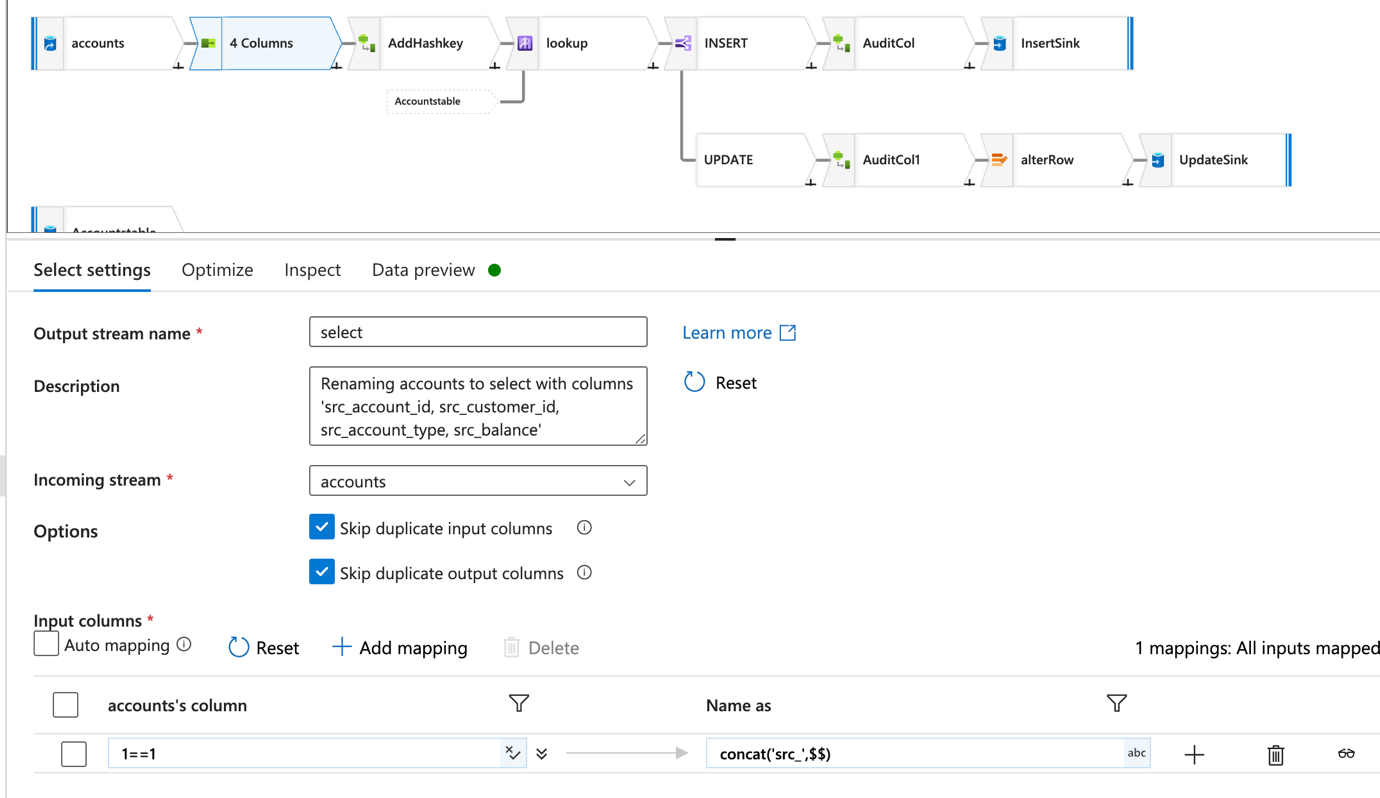
**loans**

**transactions**

* Load new records and update existing records.
* Dataflow created to take the files from silver folder.
* Source created with dataset type and linked service as Adlsgen2.
* File path to point the file in Adlasgen2.
* Imported the schema of the file.

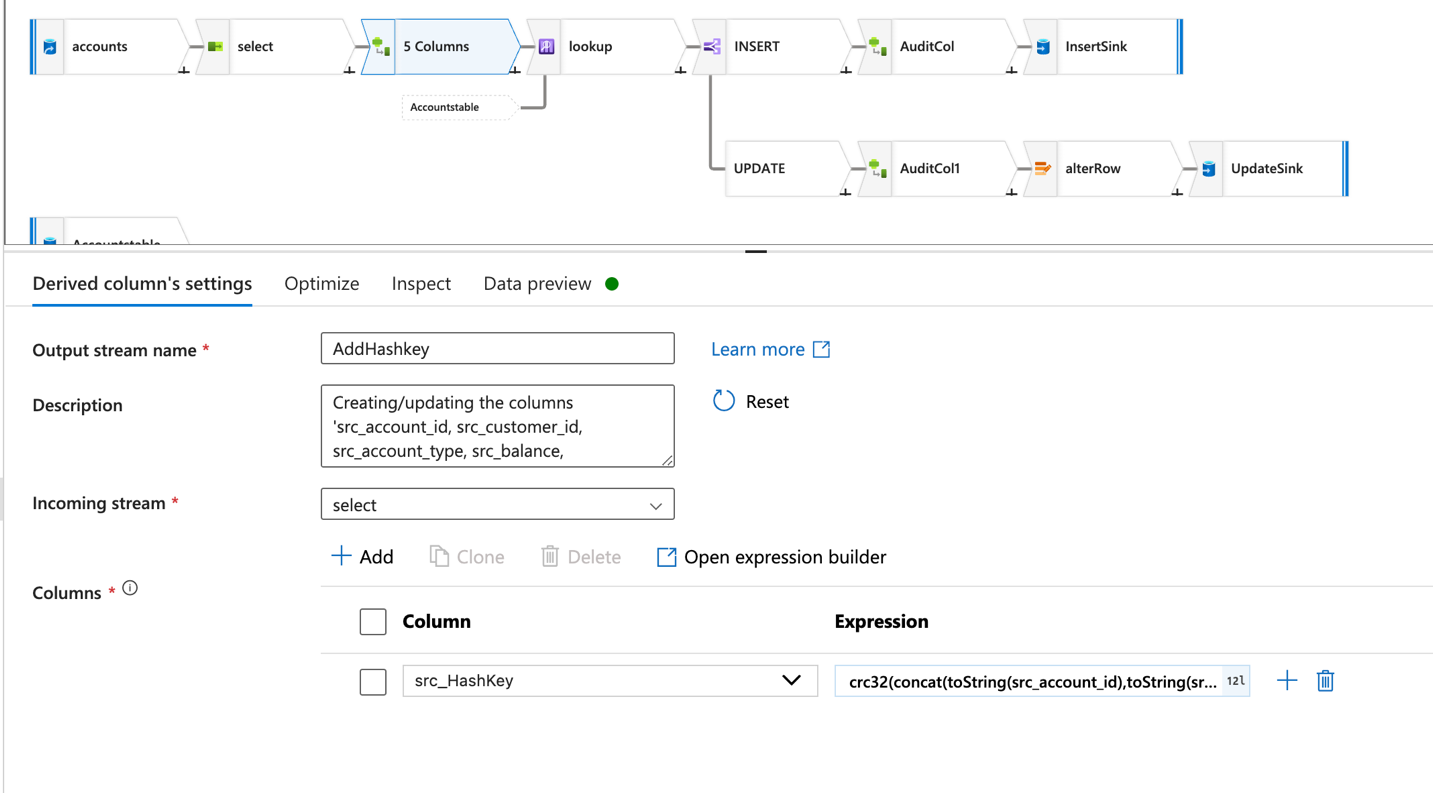


• Select transformation to Rename the column names.



• Used created hash key for comparison:

crc32(concat(toString(src\_account\_id),toString(src\_customer\_id),src\_account\_type,toString(src\_balance)))



**Target Table created with linked service Azure SQL Database:**

Schema of table created in Azure SQL database where SCD type 1 data must load.

CREATE TABLE ACCOUNT\_SCDTYPE1 (

ACCOUNT\_ID INT,

CUSTOMER\_ID INT,

ACCOUNT\_TYPE VARCHAR(50),

BALANCE DECIMAL(18,2),

CREATED\_BY VARCHAR(100),

UPDATED\_BY VARCHAR(100),

CREATE\_DATE DATETIME,

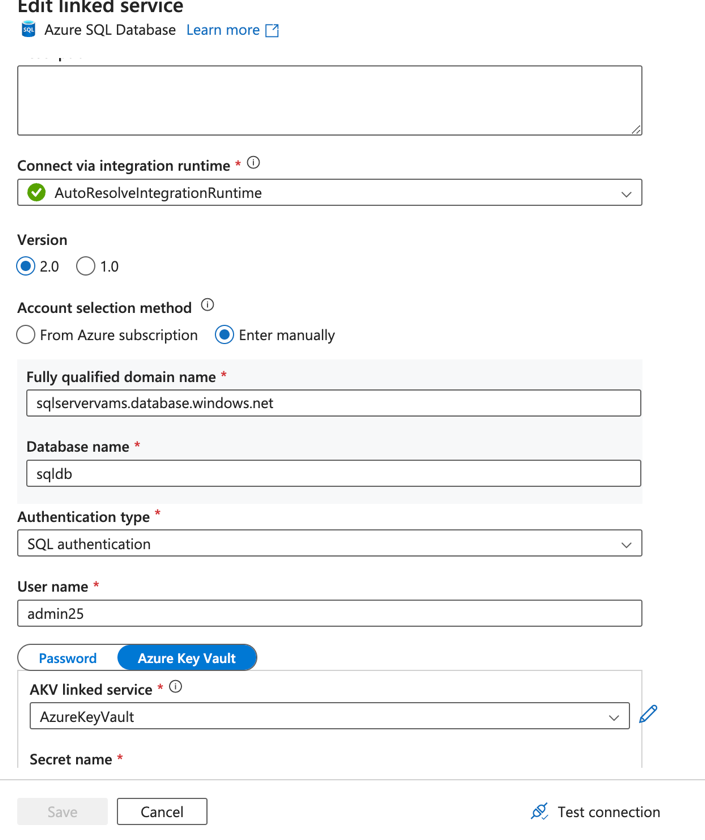
UPDATE\_DATE DATETIME,

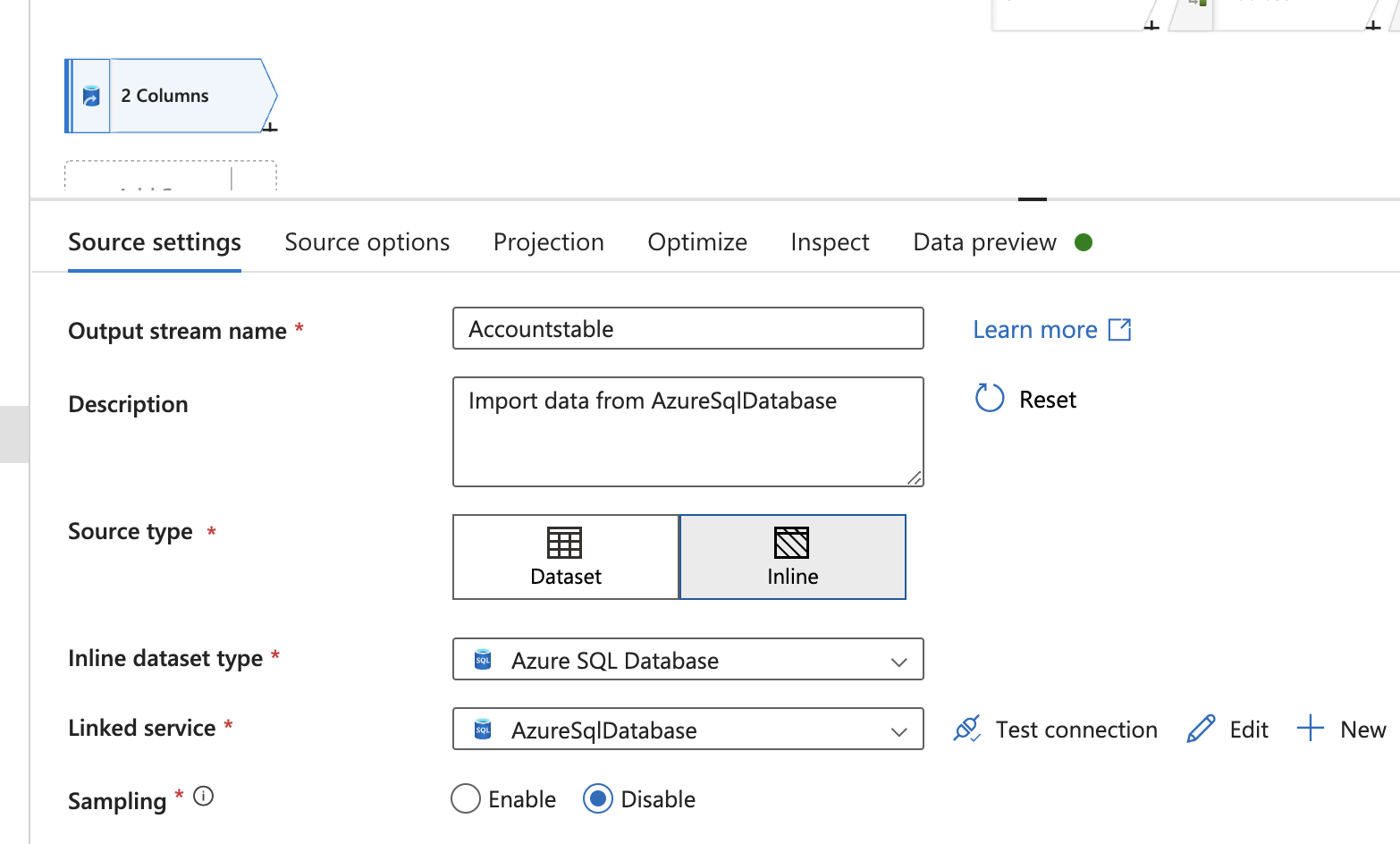
HASHKEY BIGINT

);

**Azure Key Vault:**

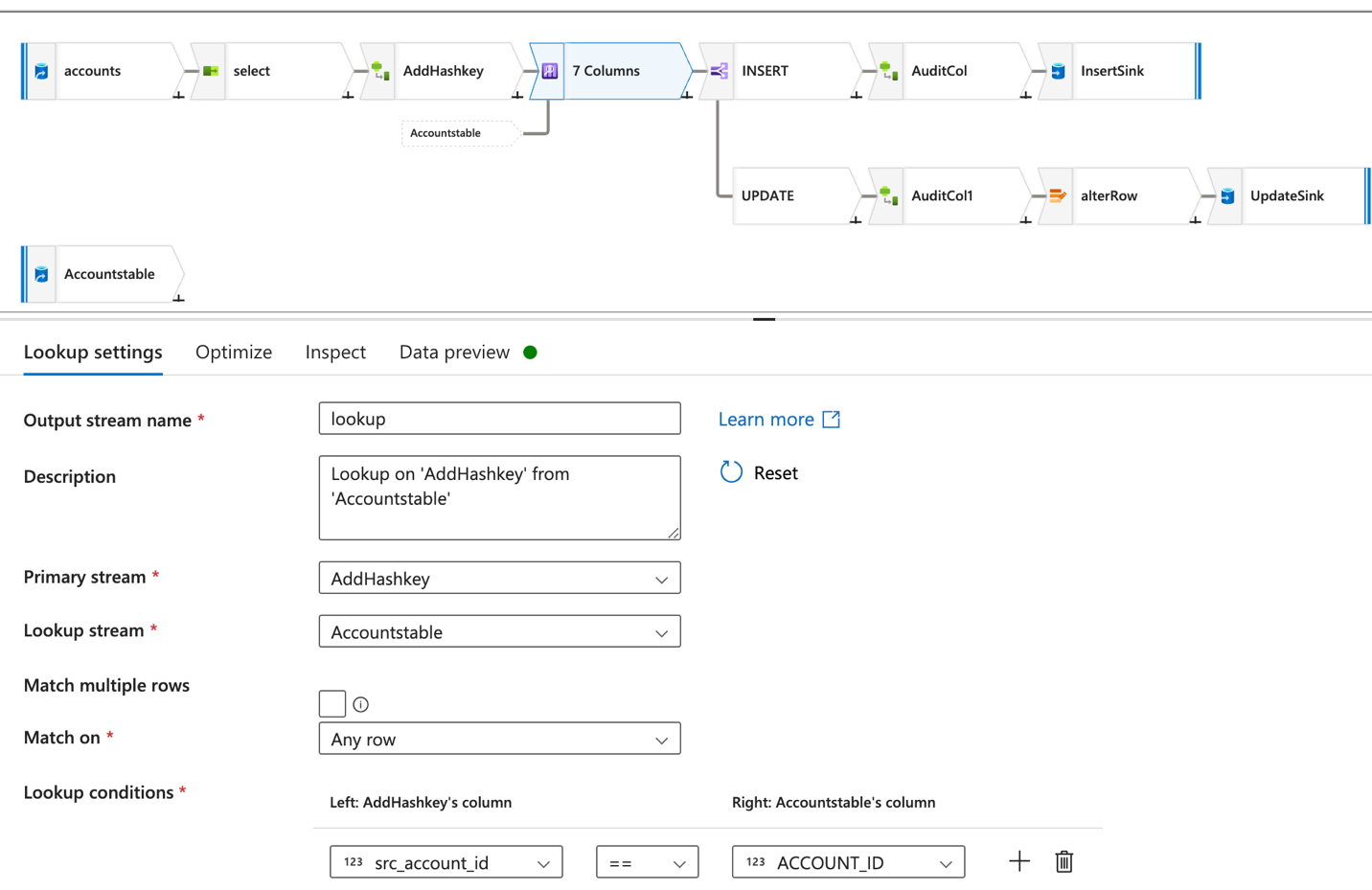
Used to securely retrieve SQL Database password for sink configuration.





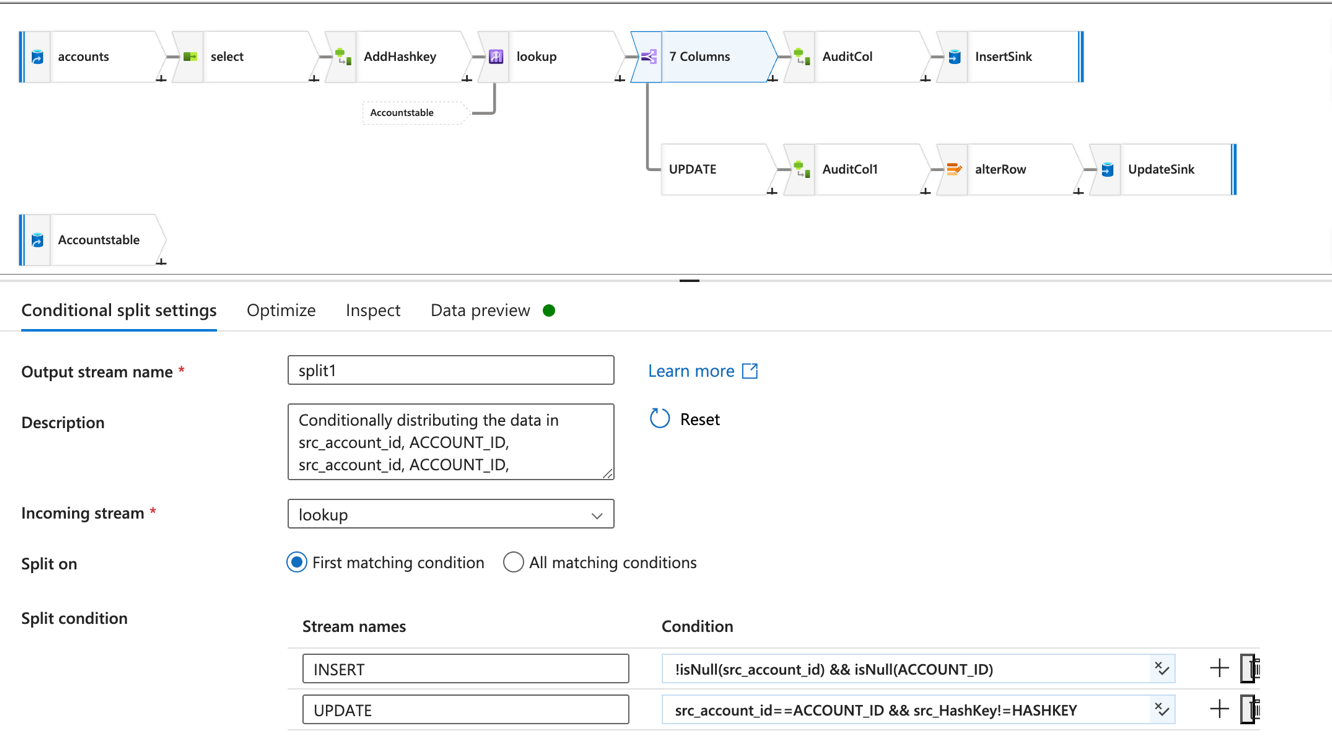
**LOOK UP to join TARGET to the Source stream.**

• Joining both the streams based on the condition.

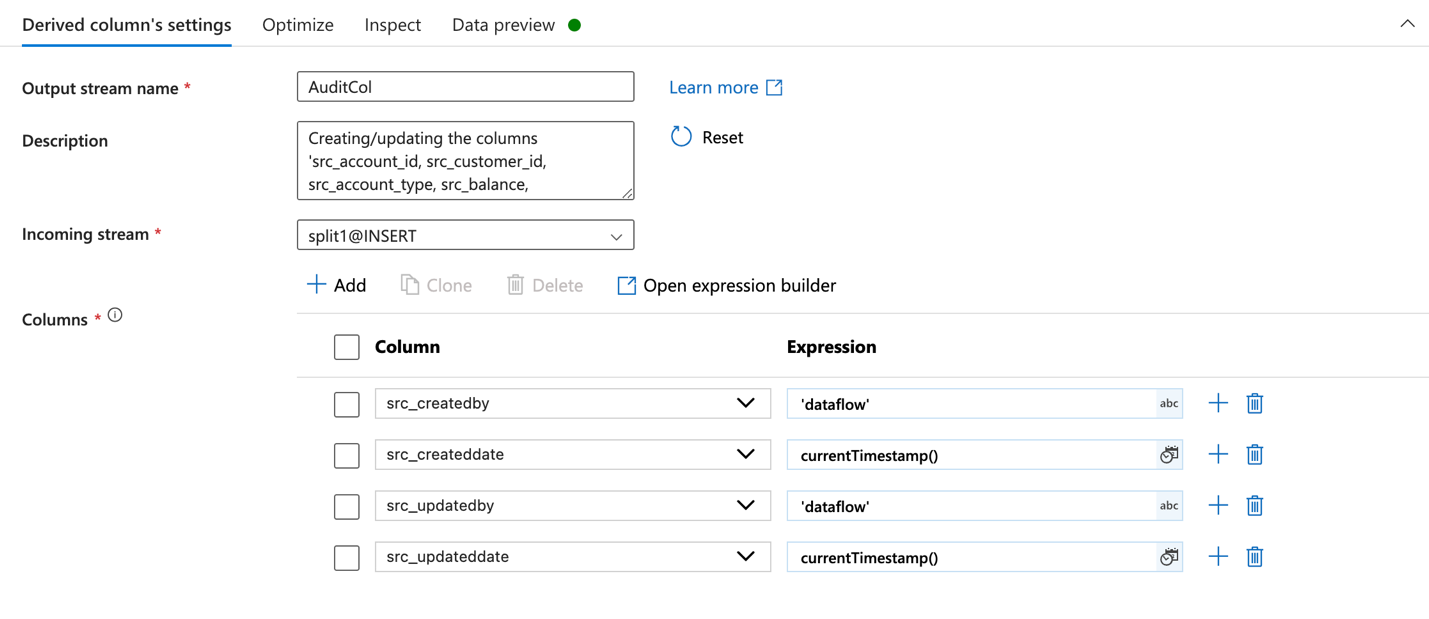


**SPLIT CONDITION:**

**Insert Stream:** Inserts new rows into the table based on the condition



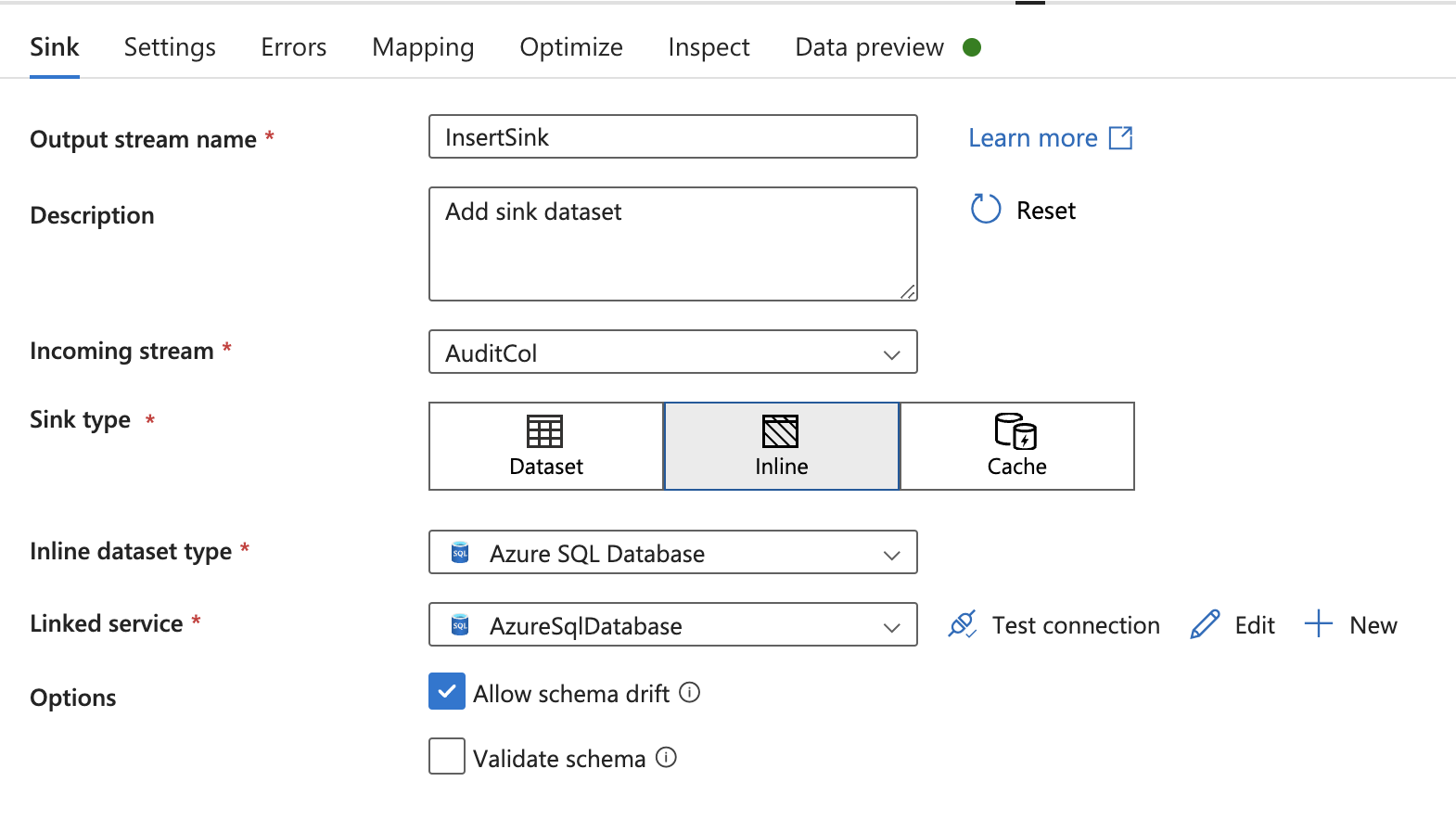
**Derived column** to create new columns to table.



**SINK** – to load the inserted rows into the target.

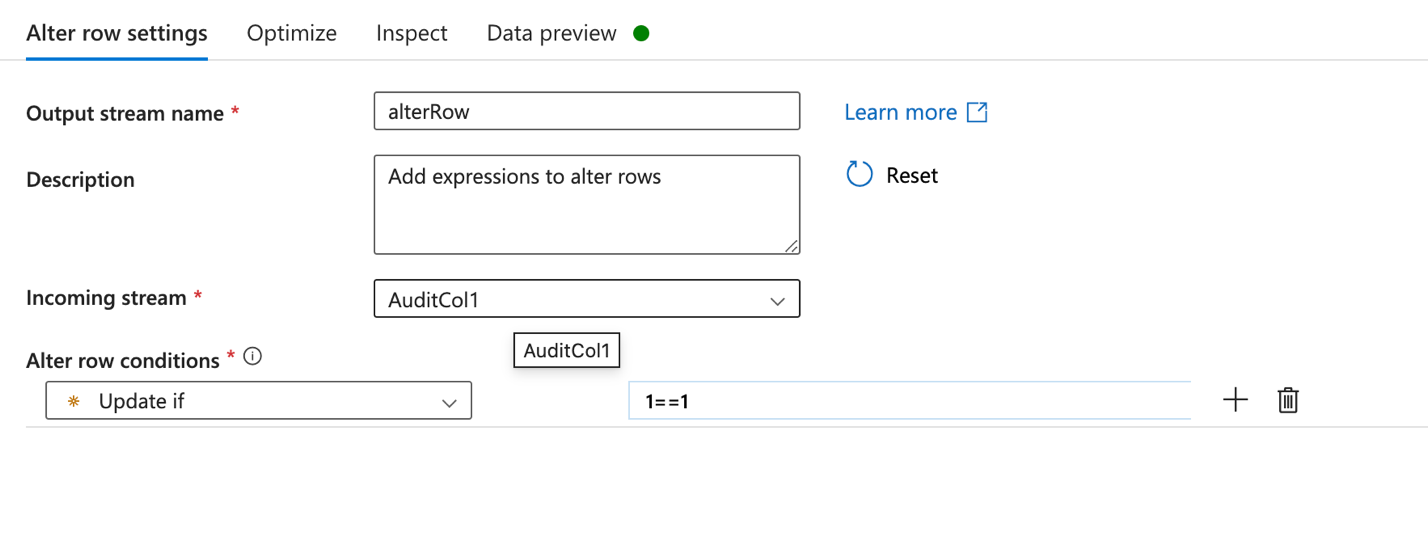
• Select schema name and table name. Select allow insert.

• Mapping input and output columns.



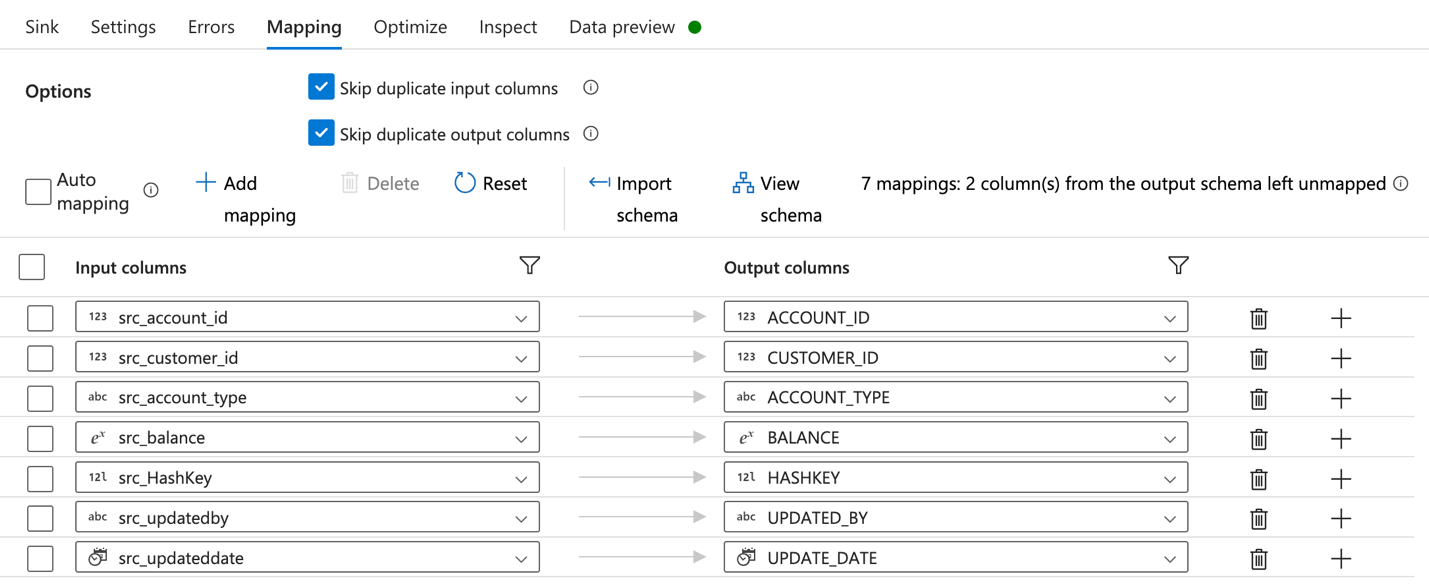
**UPDATE STREAM** - To update the modified values of the row in the table based on the condition

**Alter Row** to perform the update operation



**SINK2** - to load the updated row values into the target.

• Column mapping.



**SCD Type 2 Implemented For:**

**Customers**

* Store historical records by setting isActive = 0 for old records
* Insert new records with isActive = 1.

**(Note: - Same steps as SCD type 1 until Split condition.)**

**UPDATE STREAM:** to update the rows based on the condition

A screenshot of a computer

AI-generated content may be incorrect.

* Create new columns.
* Store historical records by setting isActive = 0 for old records.

A screenshot of a computer

AI-generated content may be incorrect.

• ALTER ROW to perform update operation.

A screenshot of a computer

AI-generated content may be incorrect.

**SINK** – to load the updated row values into the target.

• Key colums HASHKEY and CUSTOMERID.

A screenshot of a computer

AI-generated content may be incorrect.

**INSERT STREAM** – to insert the new record into the table based on condition:

A screenshot of a computer

AI-generated content may be incorrect.

* **UNION** to combine rows with Update split.

A screenshot of a computer

AI-generated content may be incorrect.

* Create new columns. Src\_isActive column is set to 1 to insert new record into table.
* Added columns: isactive, createdby, updatedby, createdate, updatedate

A screenshot of a computer

AI-generated content may be incorrect.

* **SINK** – to insert the new record into the table.

A screenshot of a computer

AI-generated content may be incorrect.

* Mapping input and output columns to load the data into Azure SQL database.

A screenshot of a computer

AI-generated content may be incorrect.

**Final outputs written to Gold layer in ADLS Gen2 or SQL DB.**

Accounts table:

A screenshot of a computer

AI-generated content may be incorrect.

Customers table:

A screenshot of a computer

AI-generated content may be incorrect.

**SCD type 1** on Accounts table: updated record by overwriting the old values and inserted latest record.

A screenshot of a computer

AI-generated content may be incorrect.

**SCD type 2** on Customer table: updated record by inserting new row and updated old record to isActive 0. It stores history of data.

A screenshot of a computer

AI-generated content may be incorrect.

**Step 4: Master Pipeline & Scheduling**

**Master Pipeline Created:**

* Used Execute Pipeline activity to orchestrate:

1. Ingestion Pipeline (Bronze)
2. Cleaning Pipeline (Silver)
3. SCD Pipeline (Gold)

**Execution Order:**

* Linked sequentially using success dependencies.

A screenshot of a computer

AI-generated content may be incorrect.

**Trigger Setup:**

* Scheduled Trigger created to run the pipeline daily.
* Ensures automated data processing end-to-end.

A screenshot of a computer screen

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Visualization with Power BI**

**Connected SQL DB to Power BI Desktop:**

* Imported required tables.
* Built dashboards using visuals like bar charts, pie charts, tables, etc.

A screenshot of a computer

AI-generated content may be incorrect.

**GIT HUB Repository:**

[**https://github.com/pvr08/Data-Pipeline-for-Customer-Account-Analysis-Project**](https://github.com/pvr08/Data-Pipeline-for-Customer-Account-Analysis-Project)