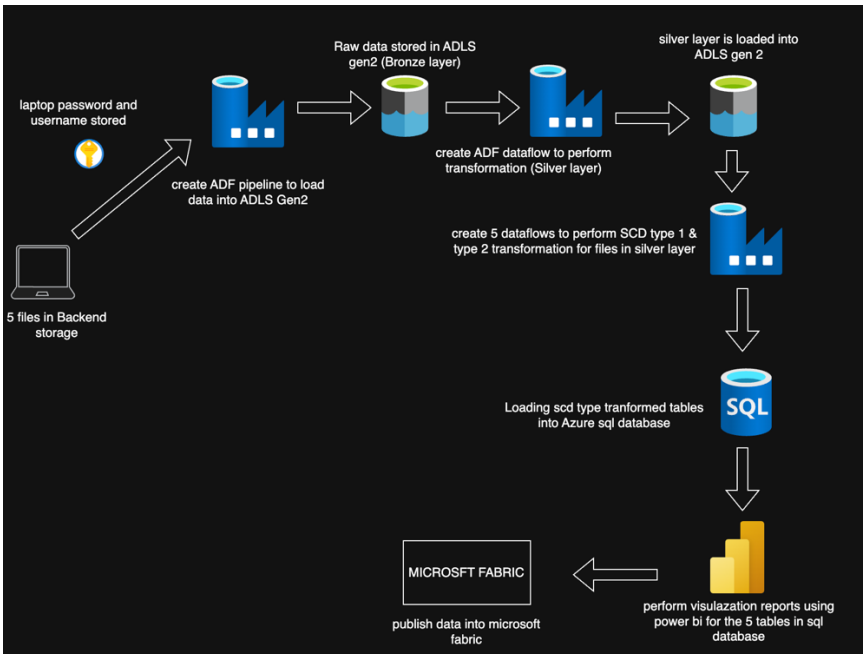
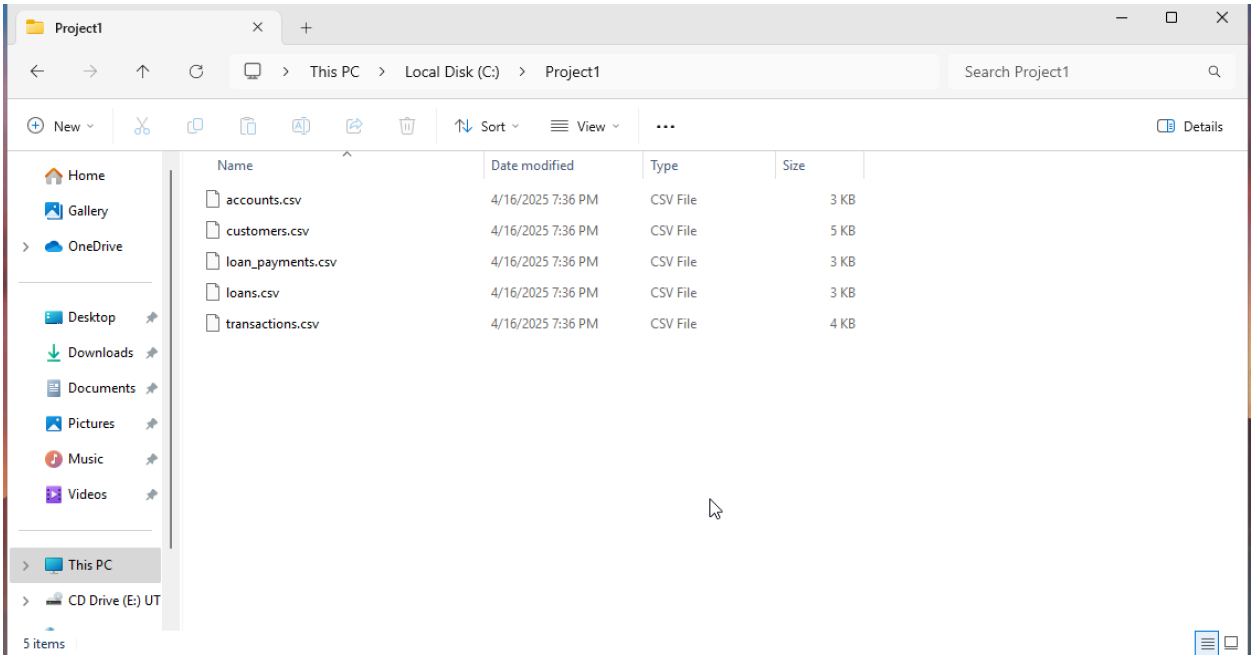
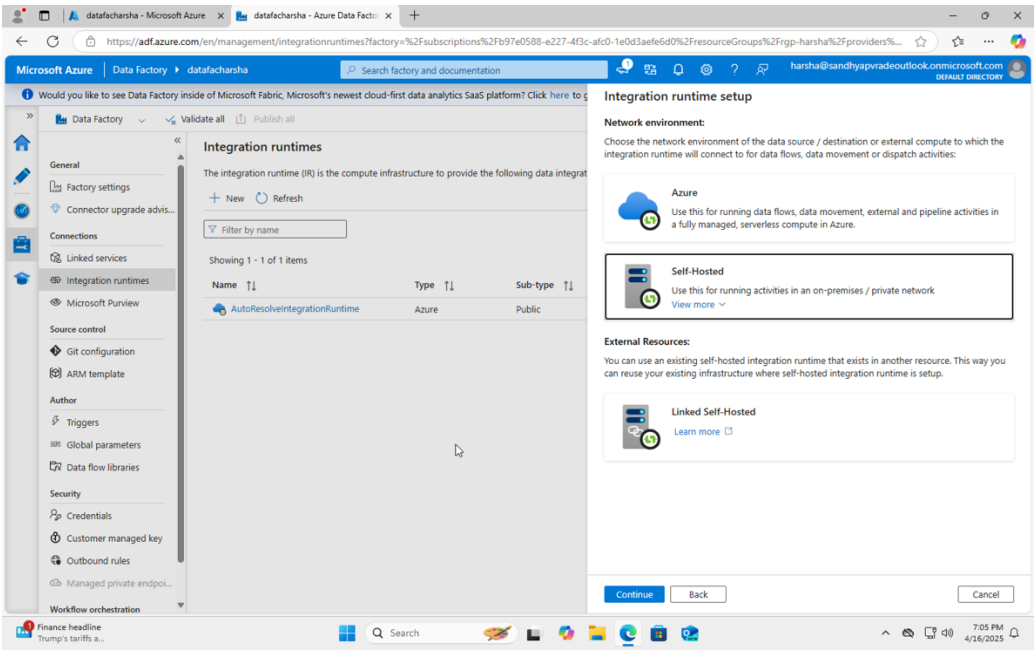


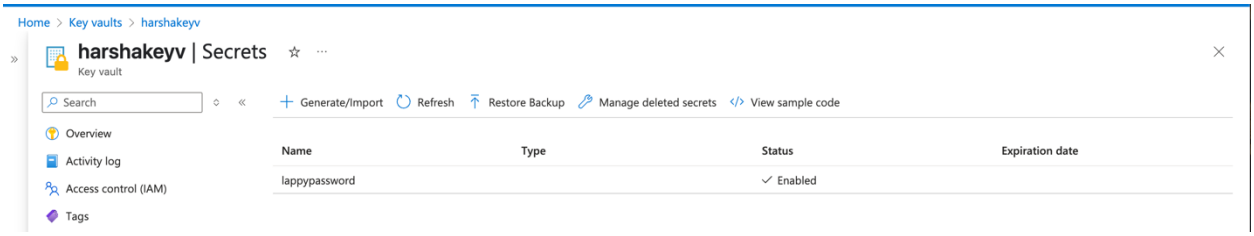
Architecture diagram for the project



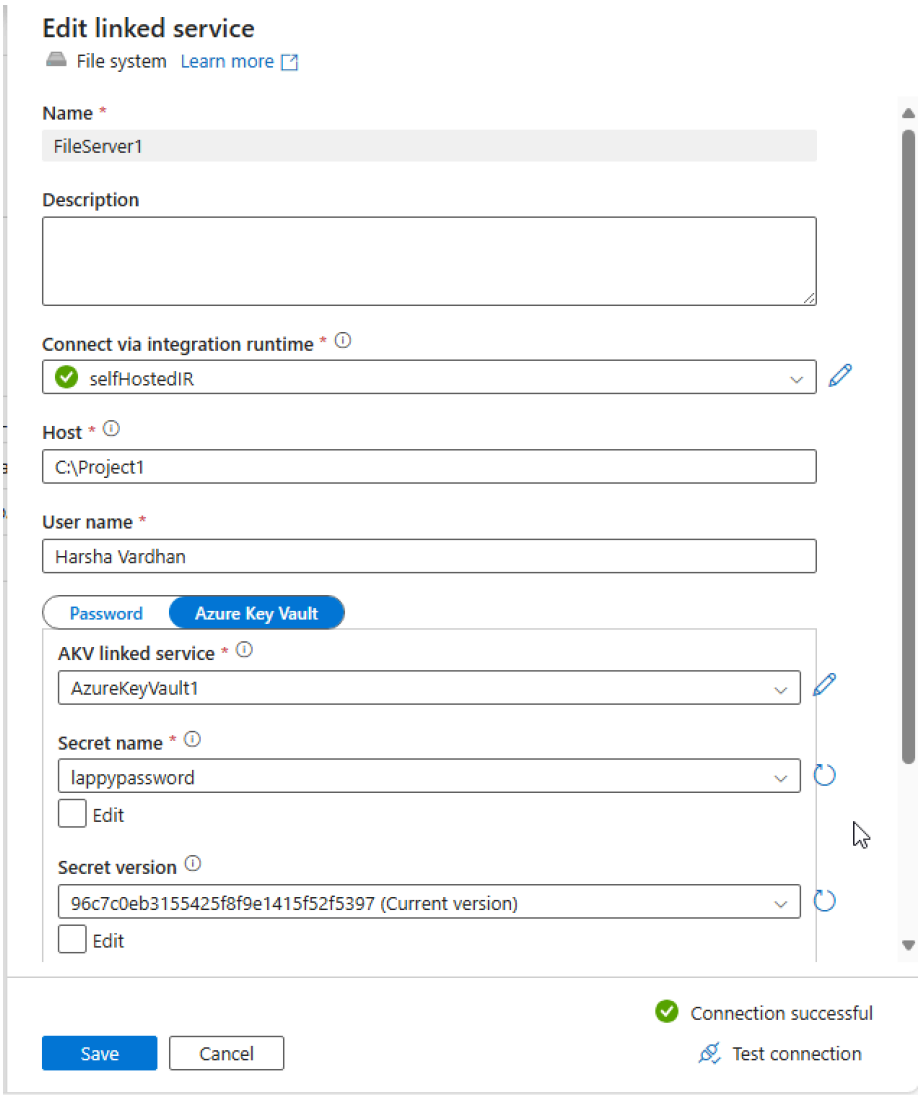
Files downloaded from kaggle



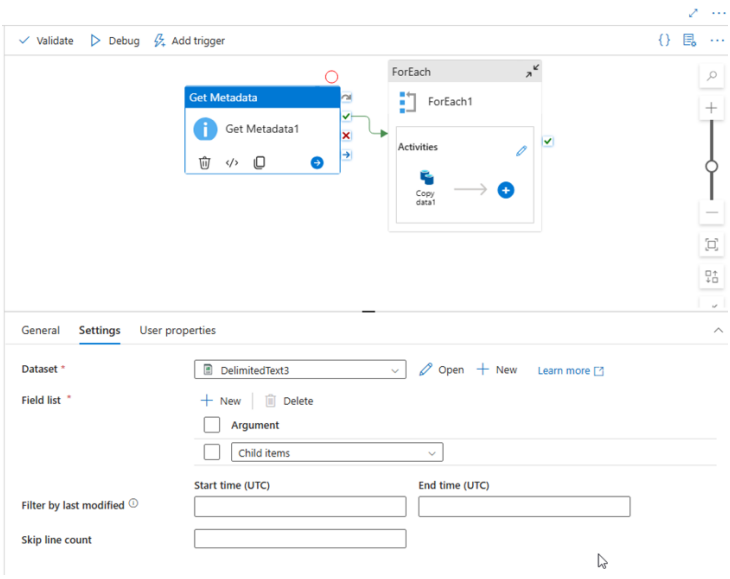
SelfHosted Integration runtime for the local machine

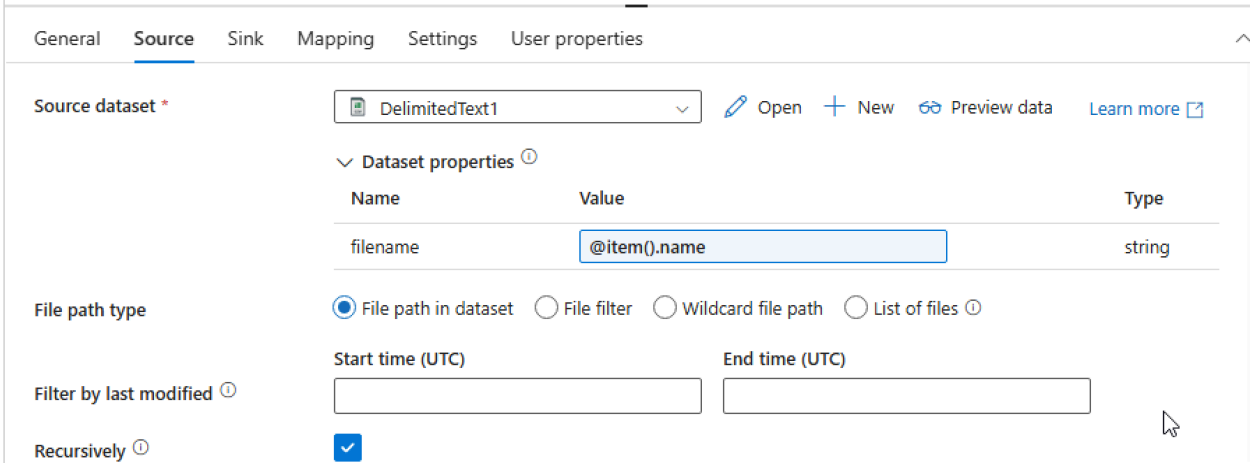


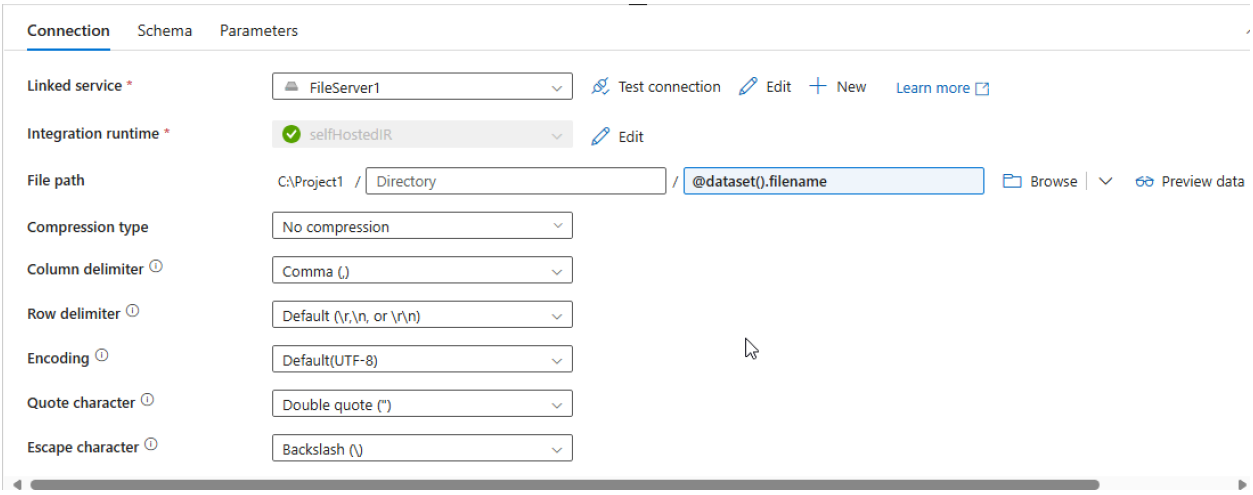
Secret key in keyvault for the laptop password

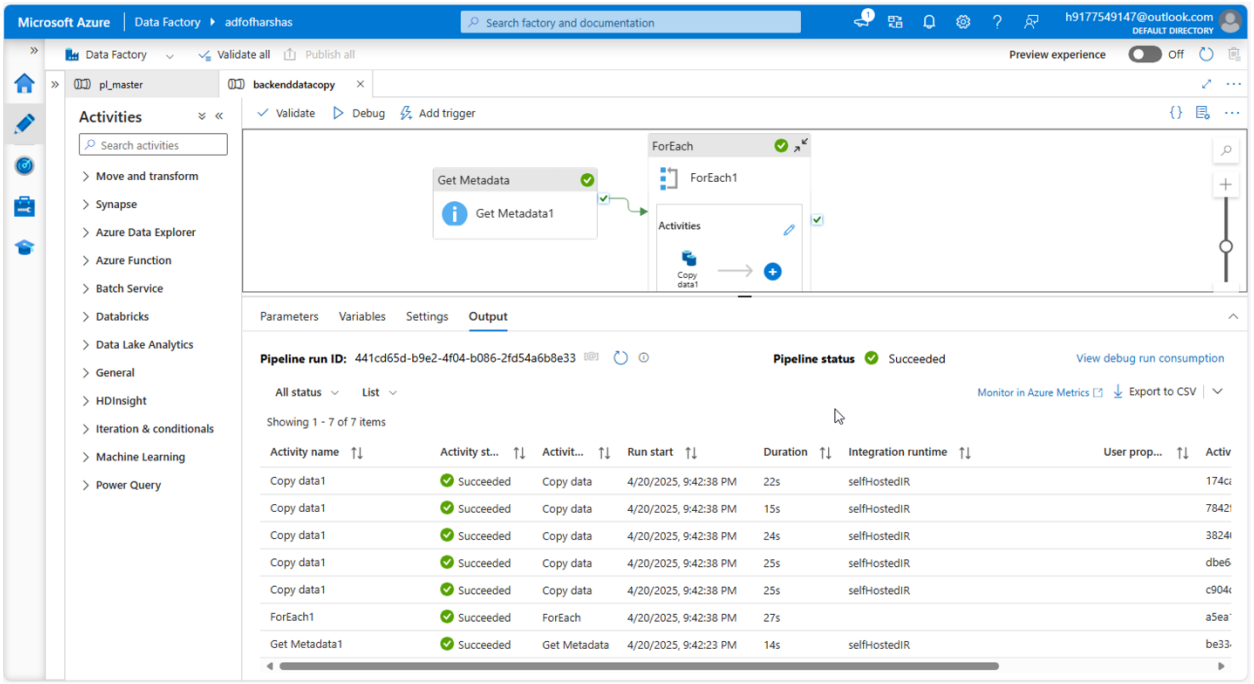


Linked service to copy the backend data

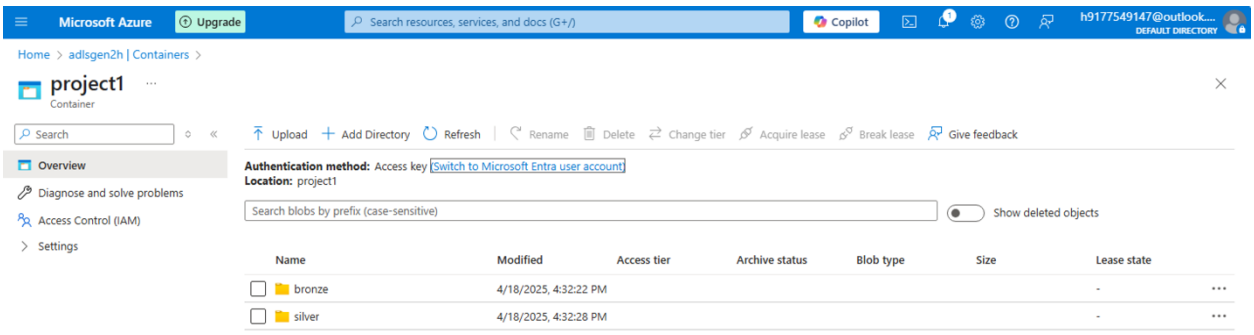

Validate
Debug
Add trigger
Get Metadata
Get Metadata1
ForEach
ForEach1
Activities
Copy data1
General
Settings
User properties
Dataset *
DelimitedText3
Open
New
Learn more
Field list *
New
Delete
Argument
Child items
Start time (UTC)
End time (UTC)
Filter by last modified
Skip line count


General
Source
Sink
Mapping
Settings
User properties
Source dataset *
DelimitedText1
Open
New
Preview data
Learn more
Dataset properties
Name
Value
Type
filename
@item().name
string
File path type
File path in dataset
File filter
Wildcard file path
List of files
Start time (UTC)
End time (UTC)
Filter by last modified
Recursively

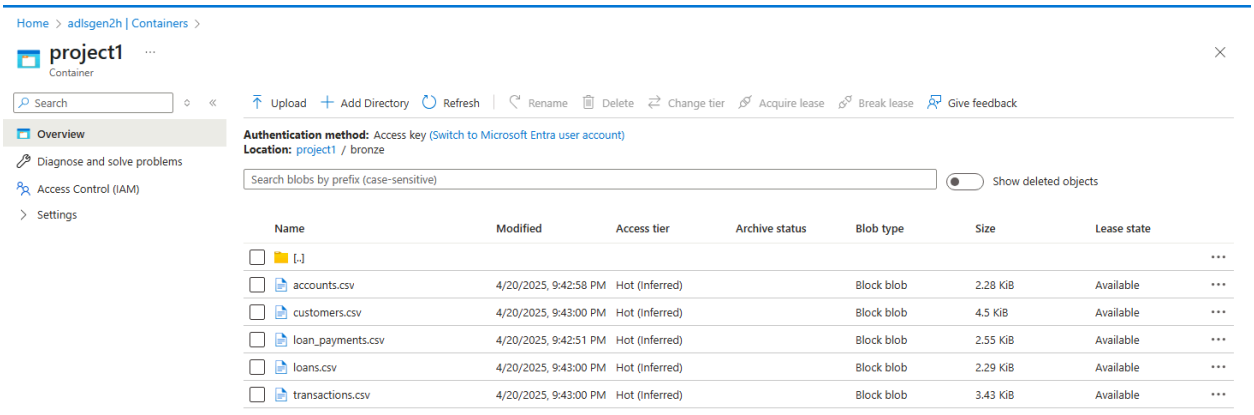

Connection
Schema
Parameters
Linked service *
FileServer1
Test connection
Edit
New
Learn more
Integration runtime *
selfHostedIR
Edit
File path
C:\Project1
Directory
@dataset().filename
Browse
Preview data
Compression type
No compression
Column delimiter
Comma (,)
Row delimiter
Default (\r,\n, or \r\n)
Encoding
Default(UTF-8)
Quote character
Double quote (")
Escape character
Backslash (\)



Pipeline to get the data from laptop to ADLS gen 2

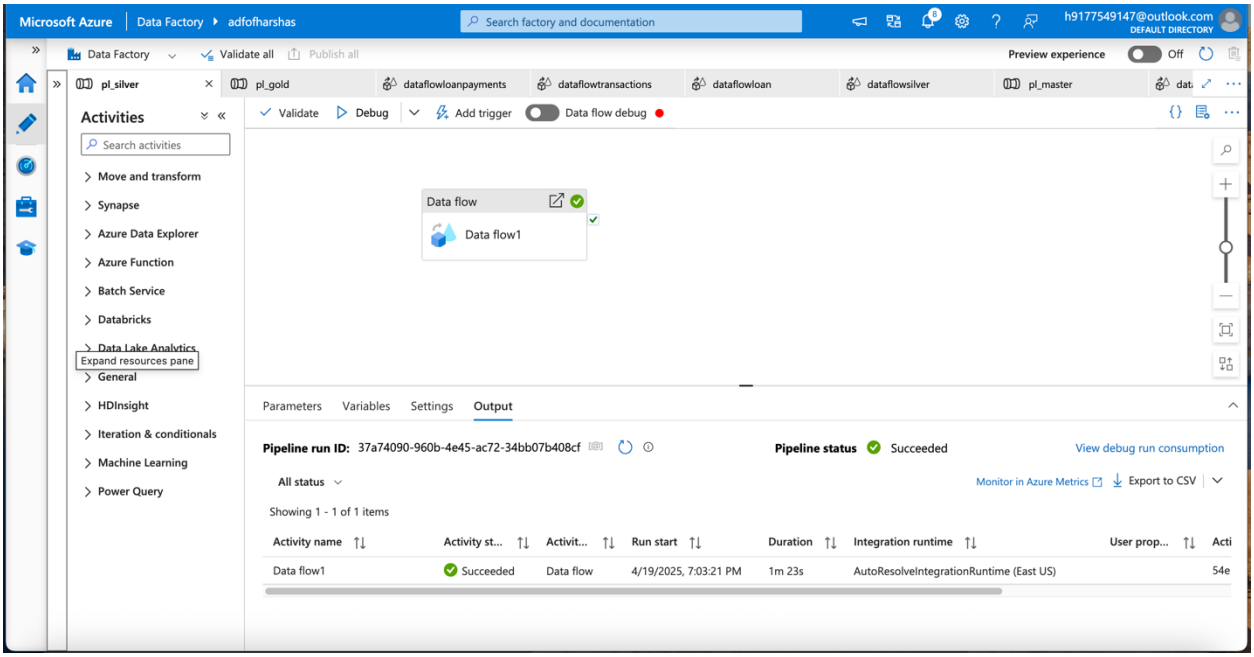


Containers created inside the ADLS to store the files

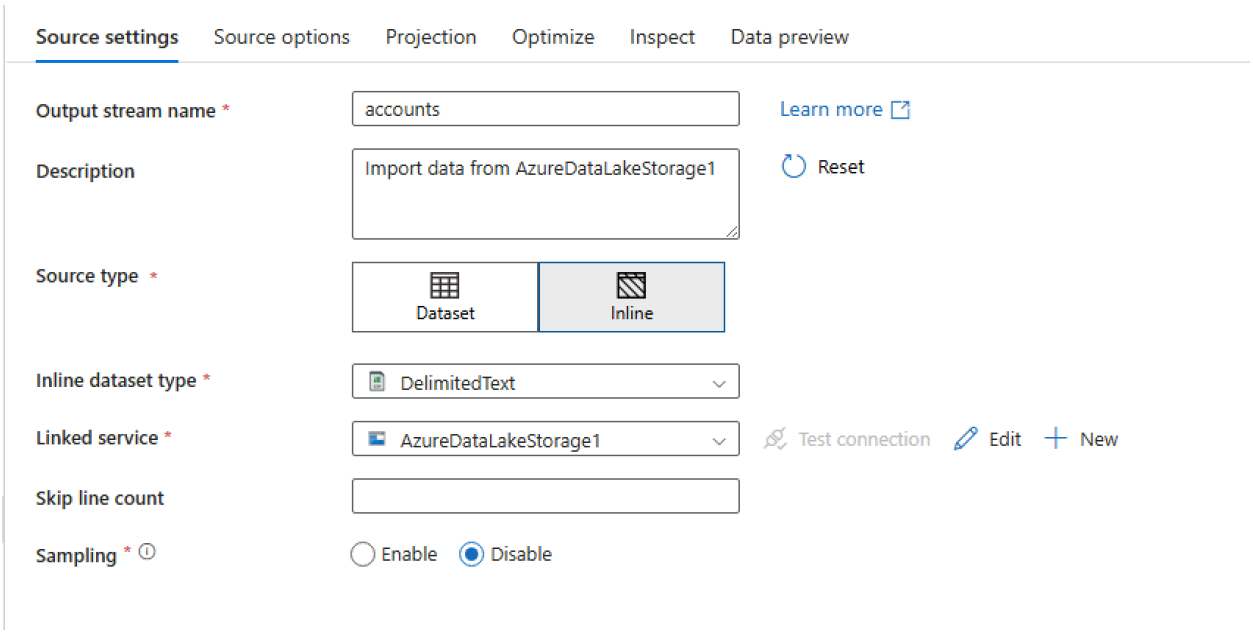


These are files loaded by successful run of the pipeline.

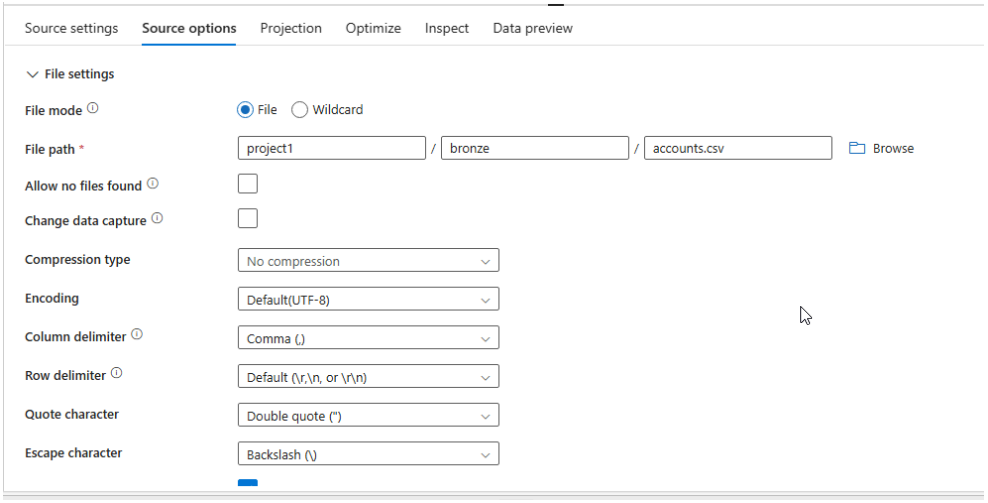
Here is the dataflow design for step_2



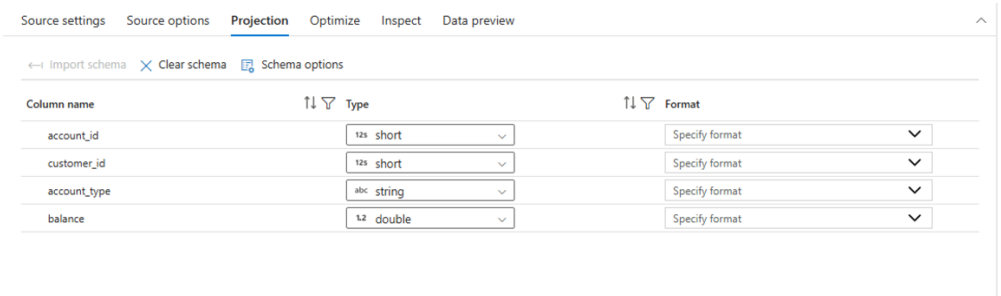
The pipeline run for step 2



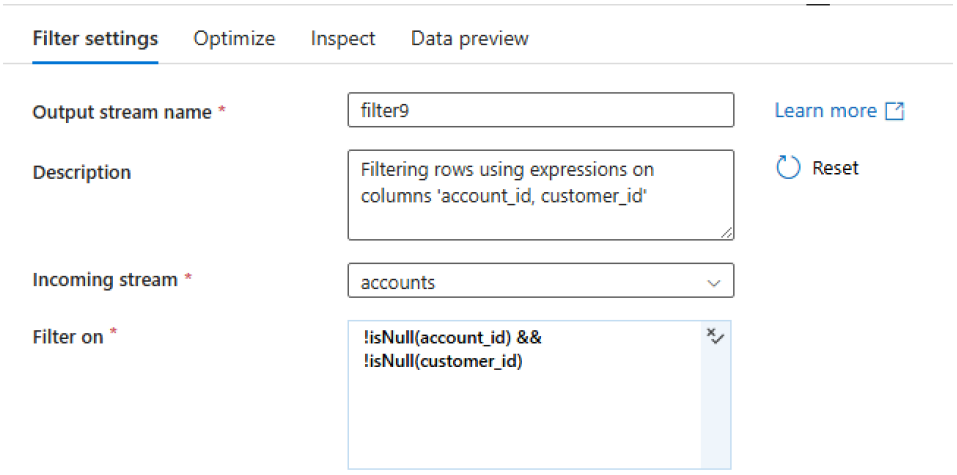
We are giving the source from our ADLS gen 2 from bronze container



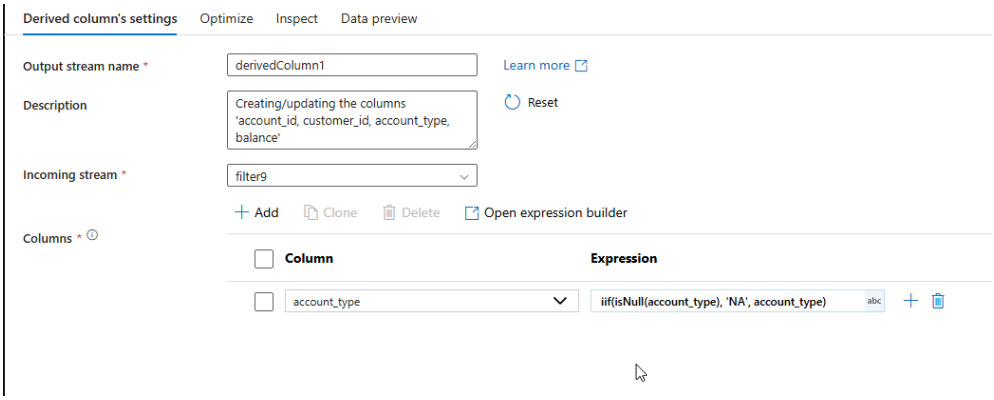
Selecting the accounts.csv file from bronze layer.



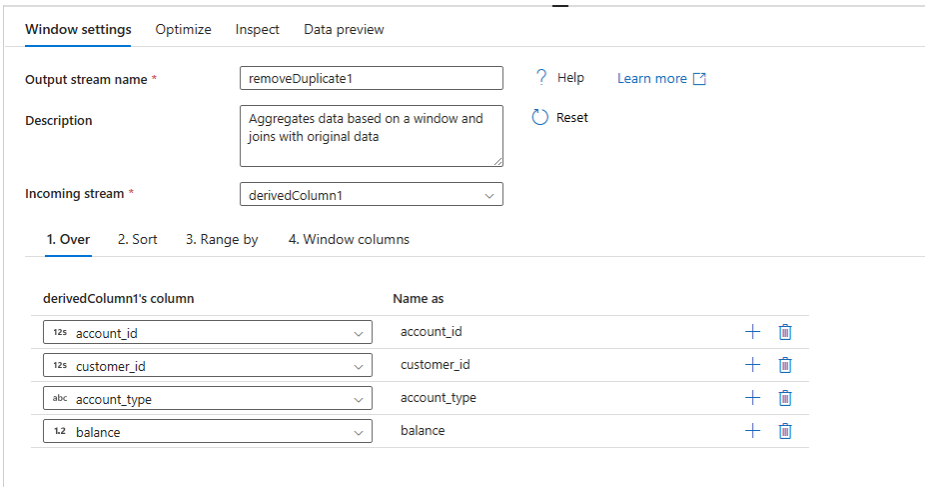
Then import schema for the file

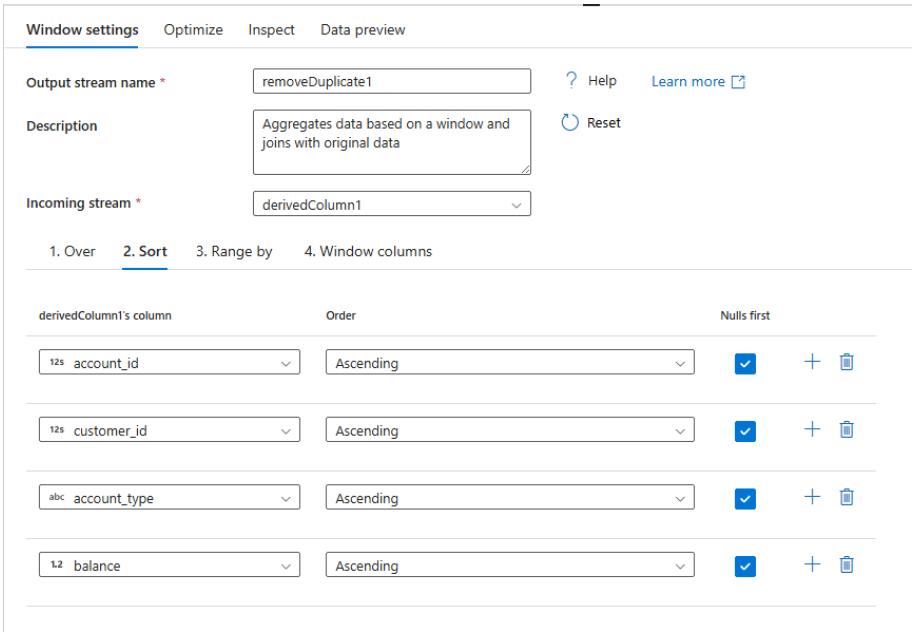


Now use filter option to check main field like account_id & customer_id if they are null we remove the entire row

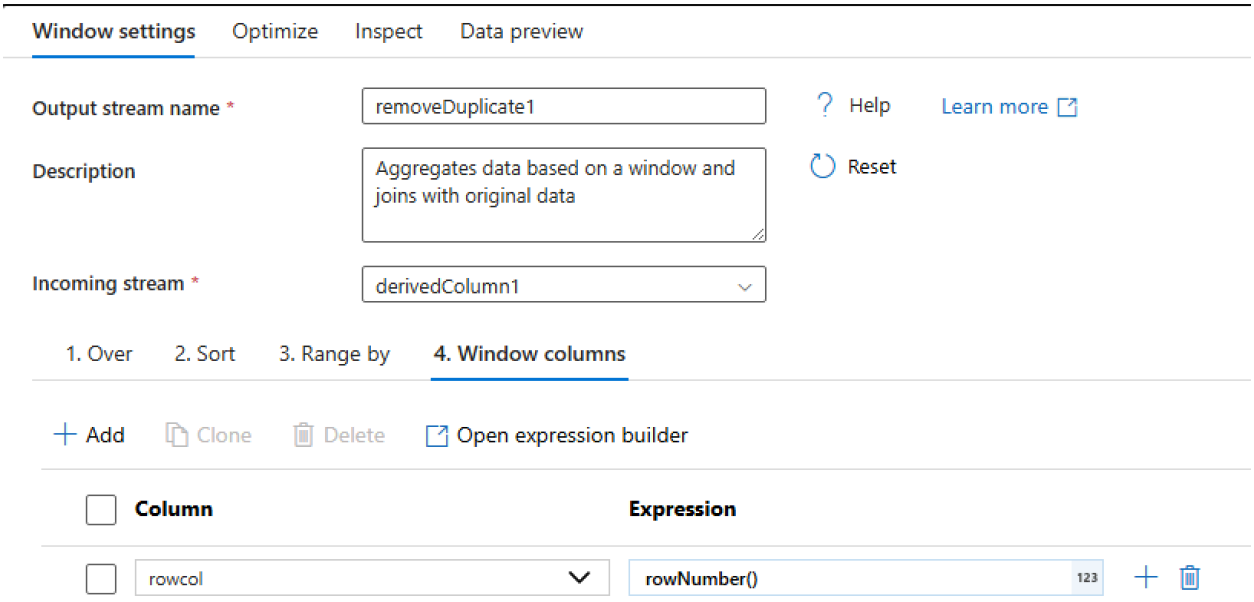


Then we give some values to the missing values in the row we are using derived column function for that with expression “iif(isNull(account_type), 'NA', account_type)”.

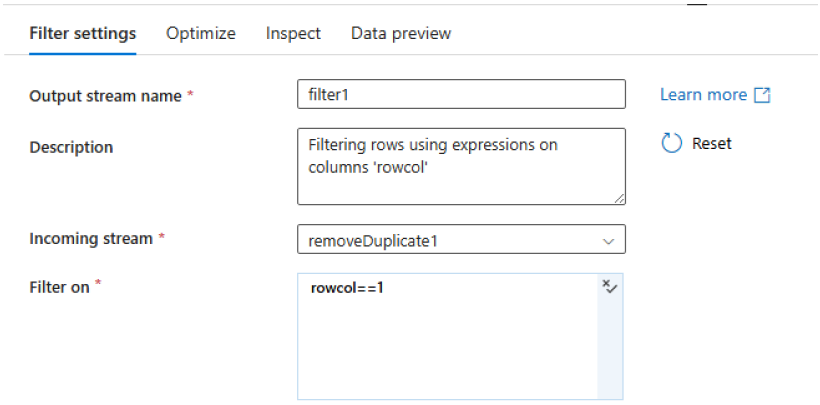




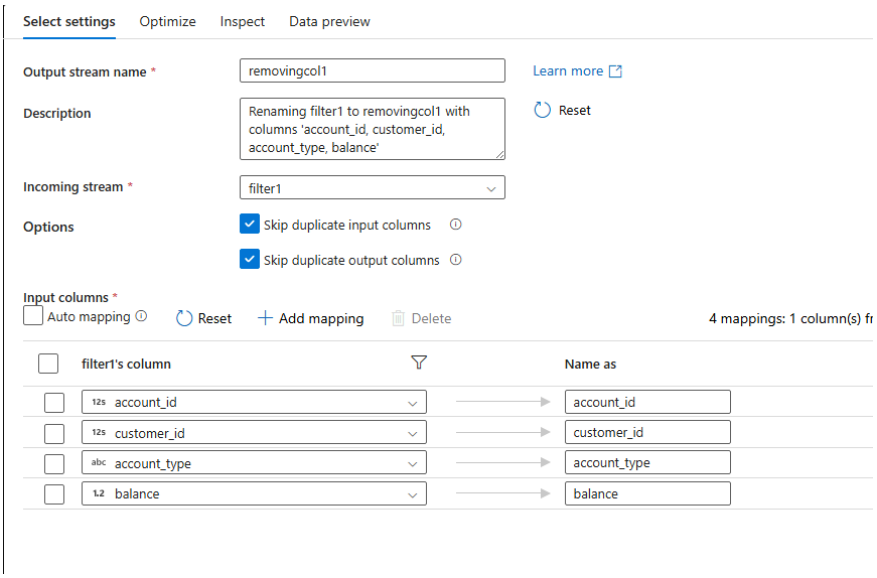
Sort in window function



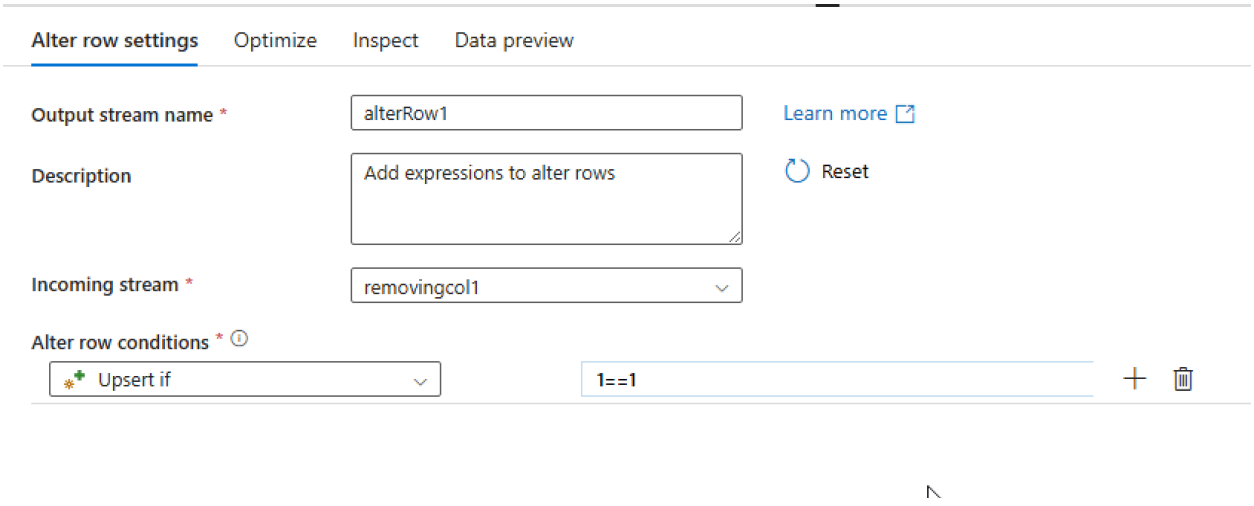
Window column expression of window function



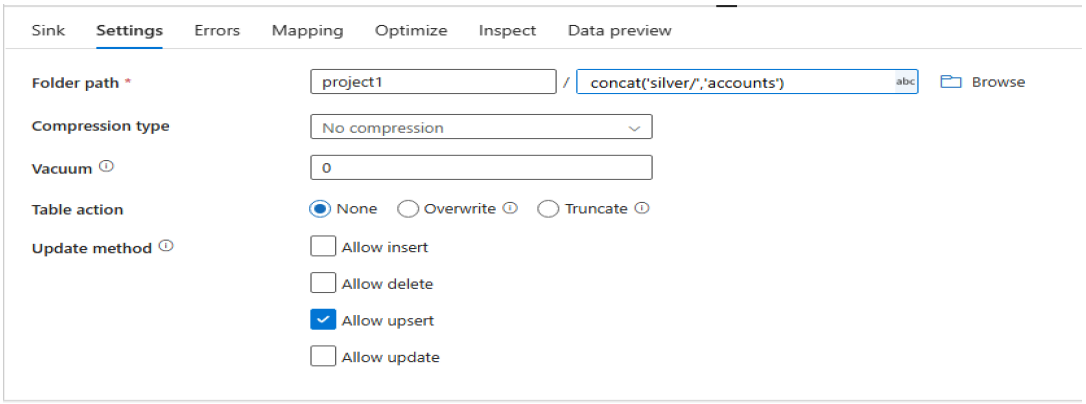
Now to remove the duplicate values we use filtering function if select rowcol==1 and only let the fields that pass-through filter.



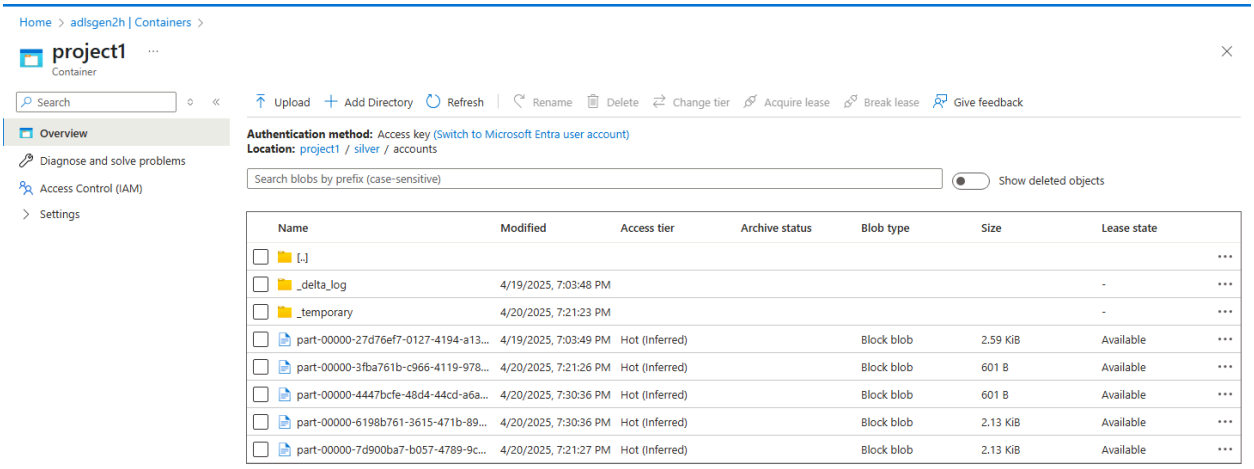
We use select function and remove rowcol and let only original rows of the table to pass

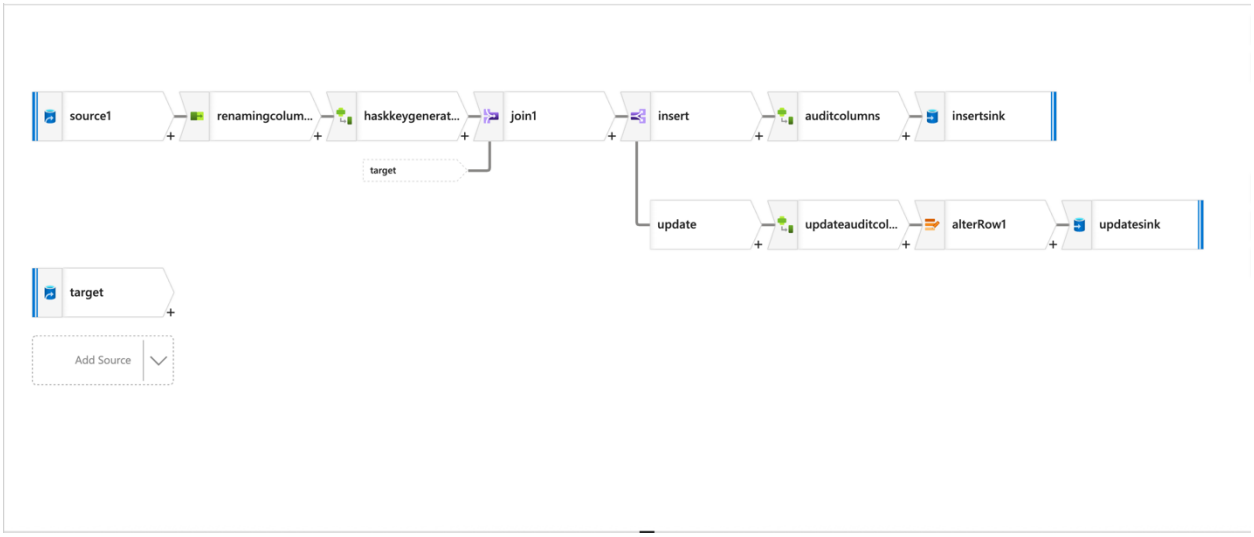


We give alter row function and give condition upsert to create delta format file.

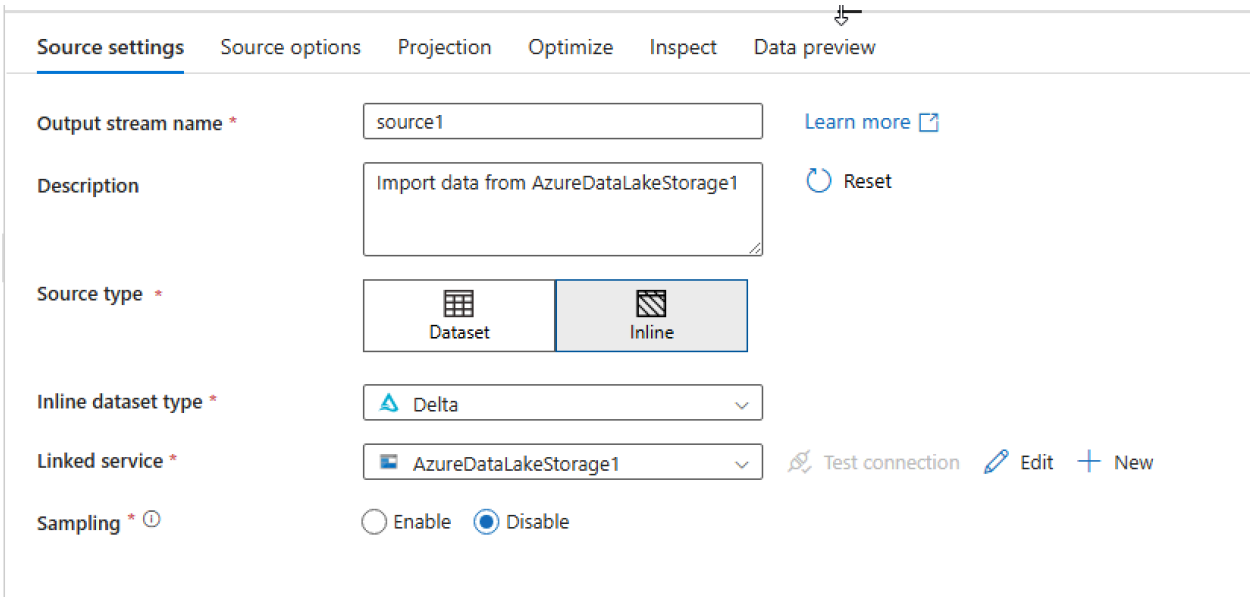


We give the path to set the file to be saved.

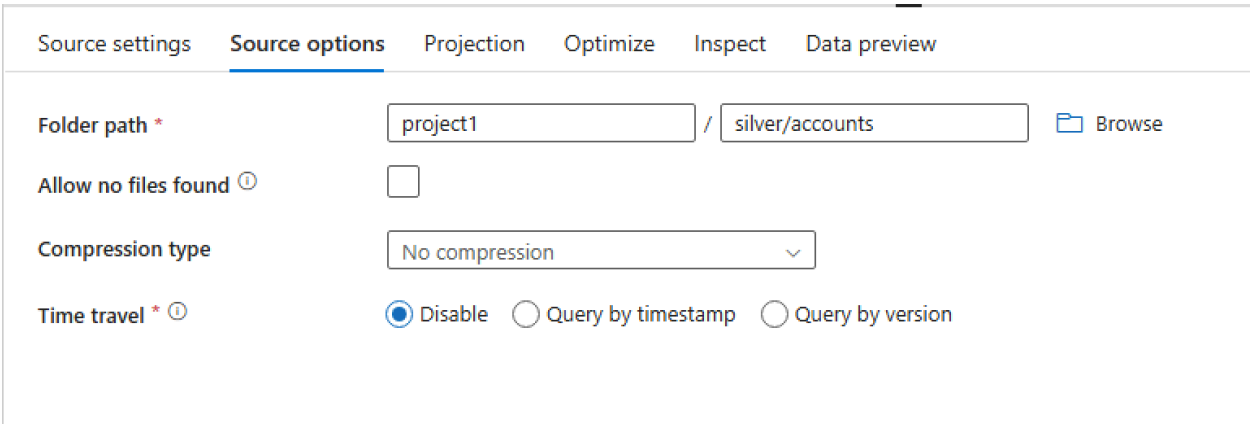




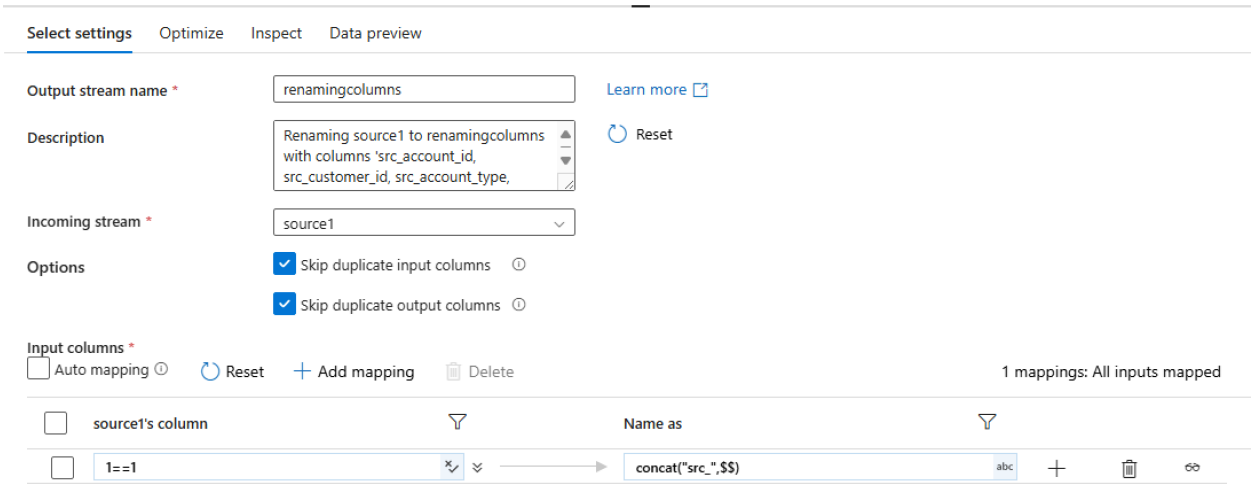
SCD type 1 transformation

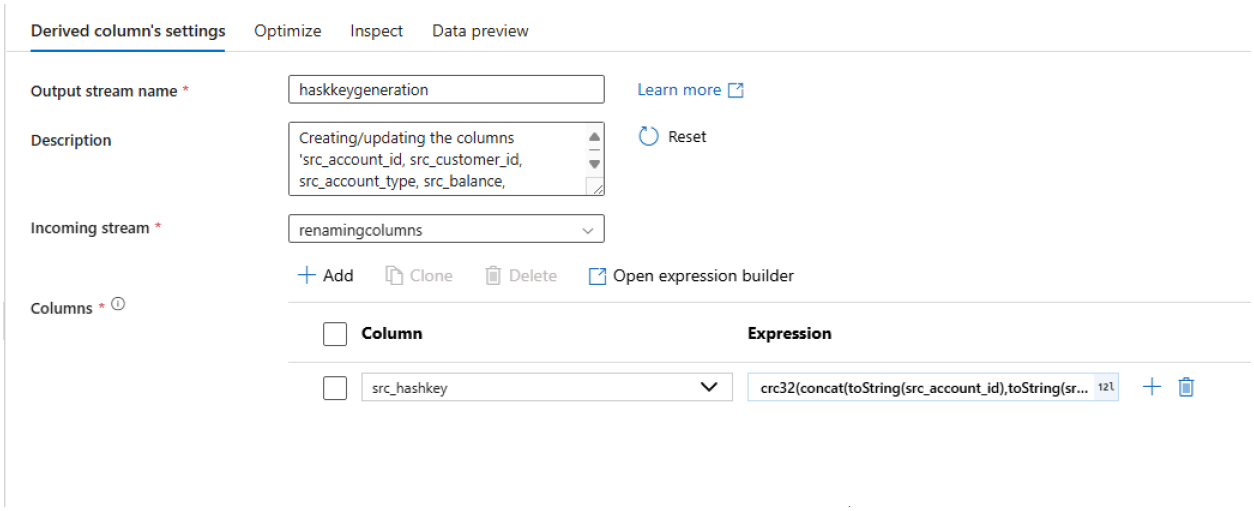


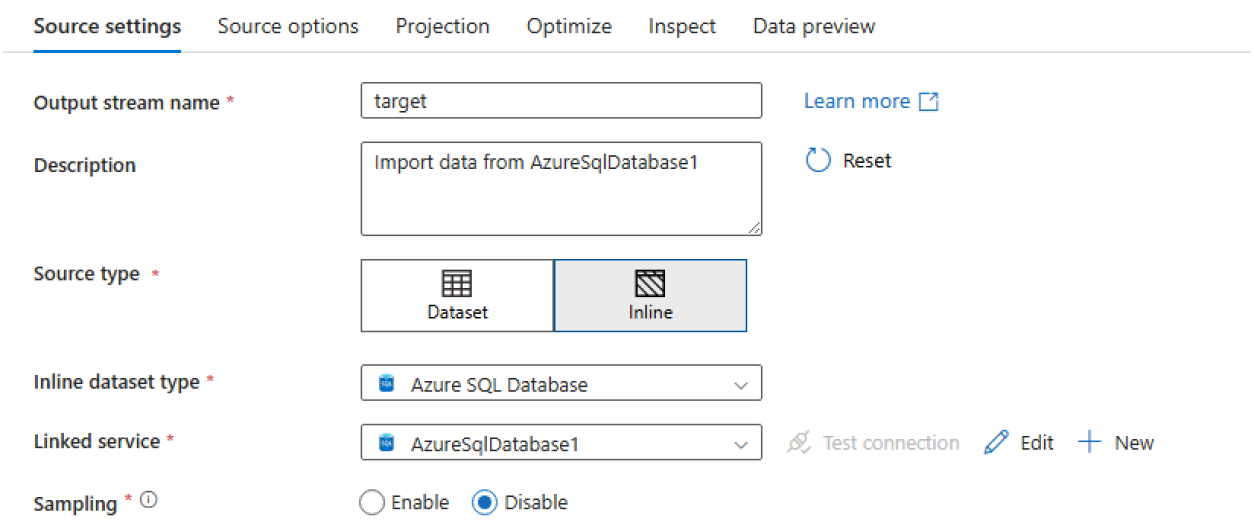
The source details are here we are selecting the delta file format from ADLS gen 2

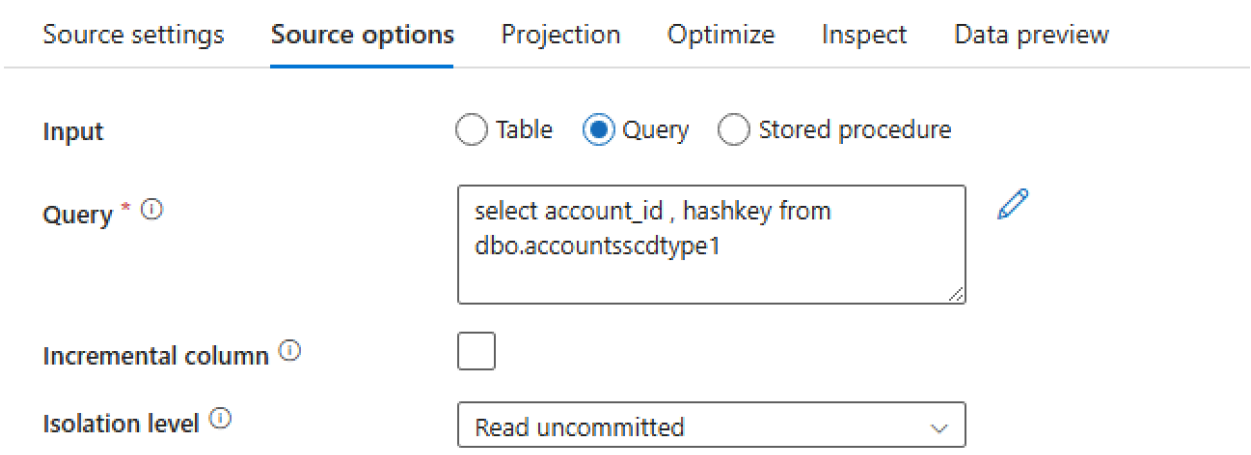


Setting the folder path

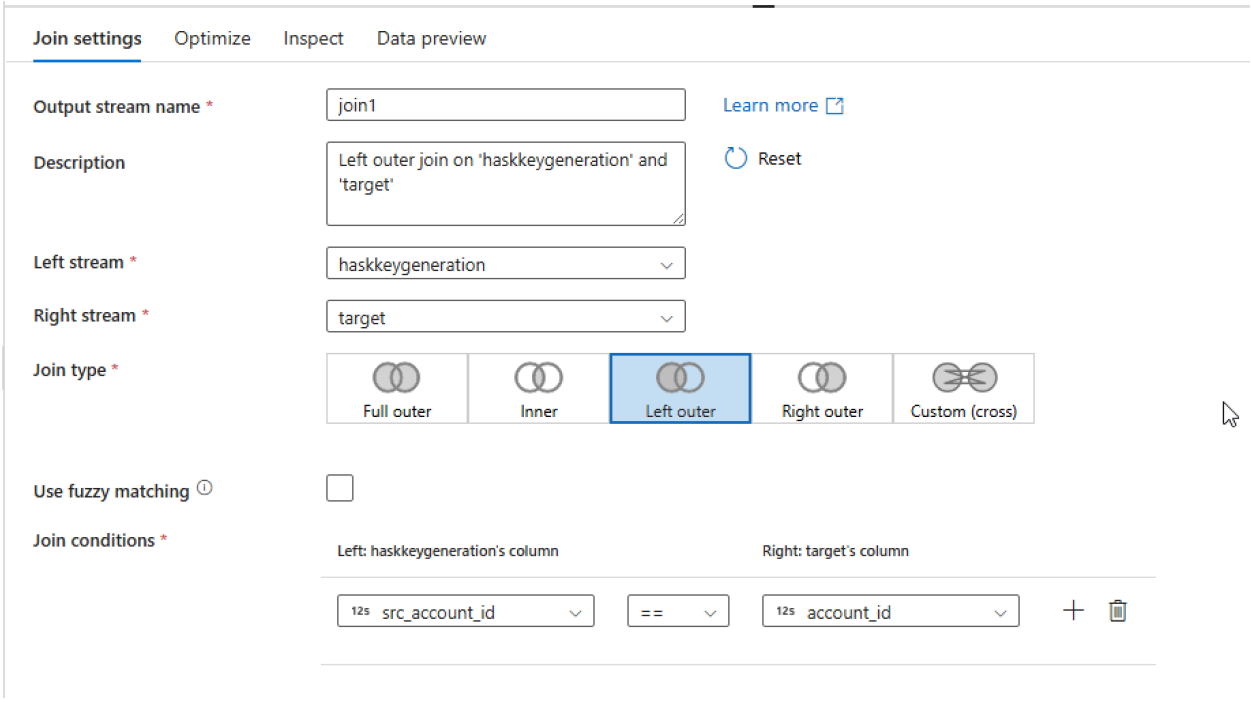

Select settings
Optimize
Inspect
Data preview
Output stream name *
renamingcolumns
Learn more
Description
Renaming source1 to renamingcolumns with columns 'src_account_id, src_customer_id, src_account_type,
Reset
Incoming stream *
source1
Options
Skip duplicate input columns
Skip duplicate output columns
Input columns *
Auto mapping
Reset
Add mapping
Delete
1 mappings: All inputs mapped
source1's column
Name as
1==1
concat("src_",$$)


Derived column's settings
Optimize
Inspect
Data preview
Output stream name *
haskkeygeneration
Learn more
Description
Creating/updating the columns 'src_account_id, src_customer_id, src_account_type, src_balance,
Reset
Incoming stream *
renamingcolumns
Add
Clone
Delete
Open expression builder
Columns *
Column
Expression
src_hashkey
crc32(concat(toString(src_account_id),toString(sr...

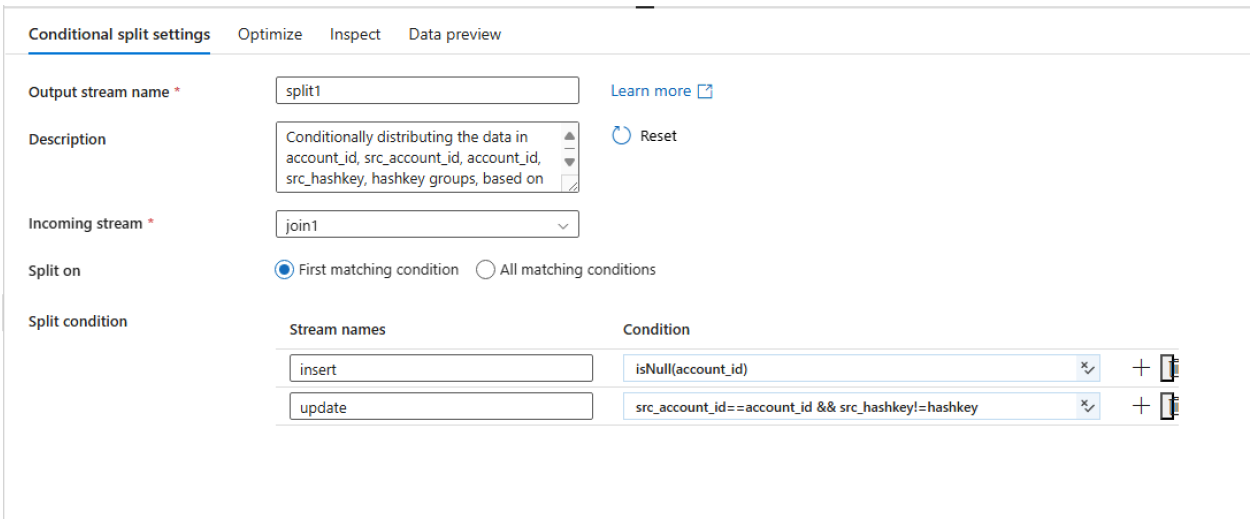

Source settings
Source options
Projection
Optimize
Inspect
Data preview
Output stream name *
target
Learn more
Description
Import data from AzureSqlDatabase1
Reset
Source type *
Dataset
Inline
Inline dataset type *
Azure SQL Database
Linked service *
AzureSqlDatabase1
Test connection
Edit
New
Sampling *
Enable
Disable



We give this query to select the account_id & haskey from the table.



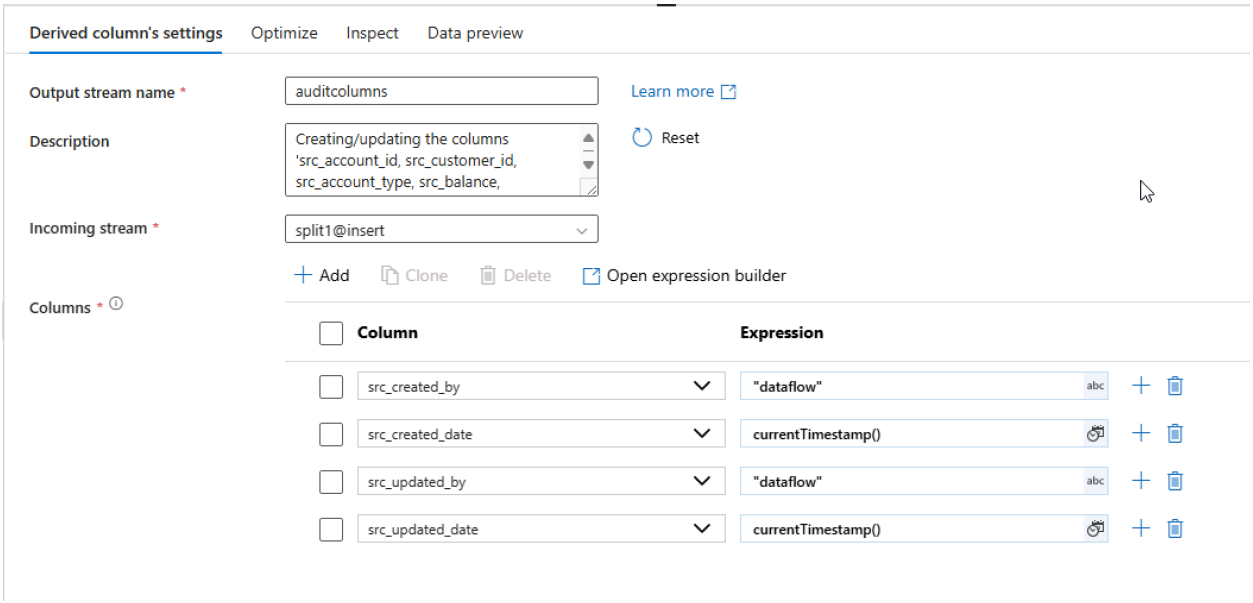
We are using left join to join the source and target



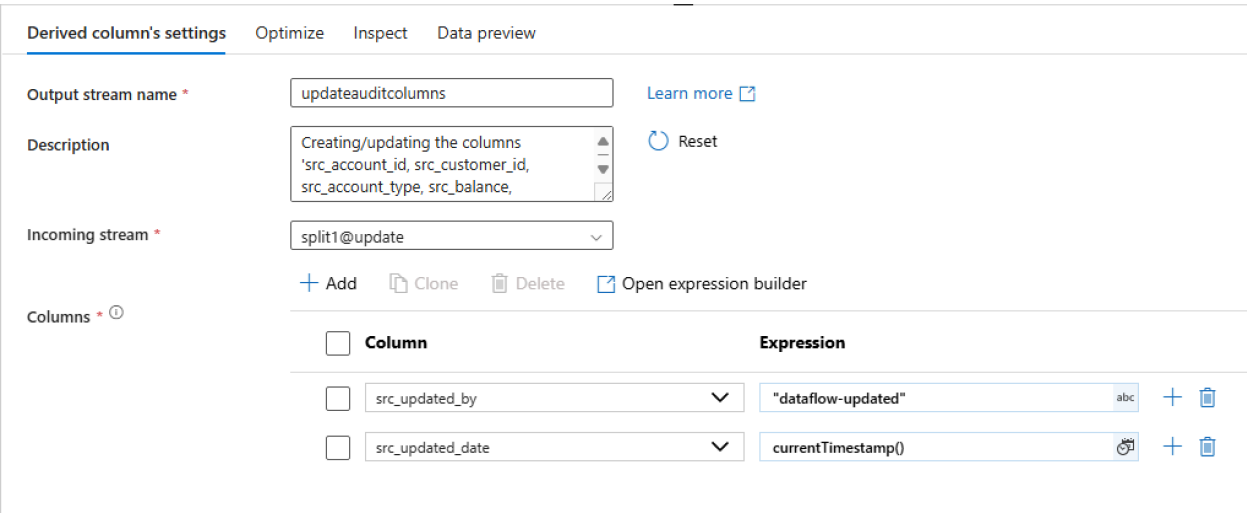
We are giving the conditions to set the insert and update functions

For insert we are giving : isnull(account_id)

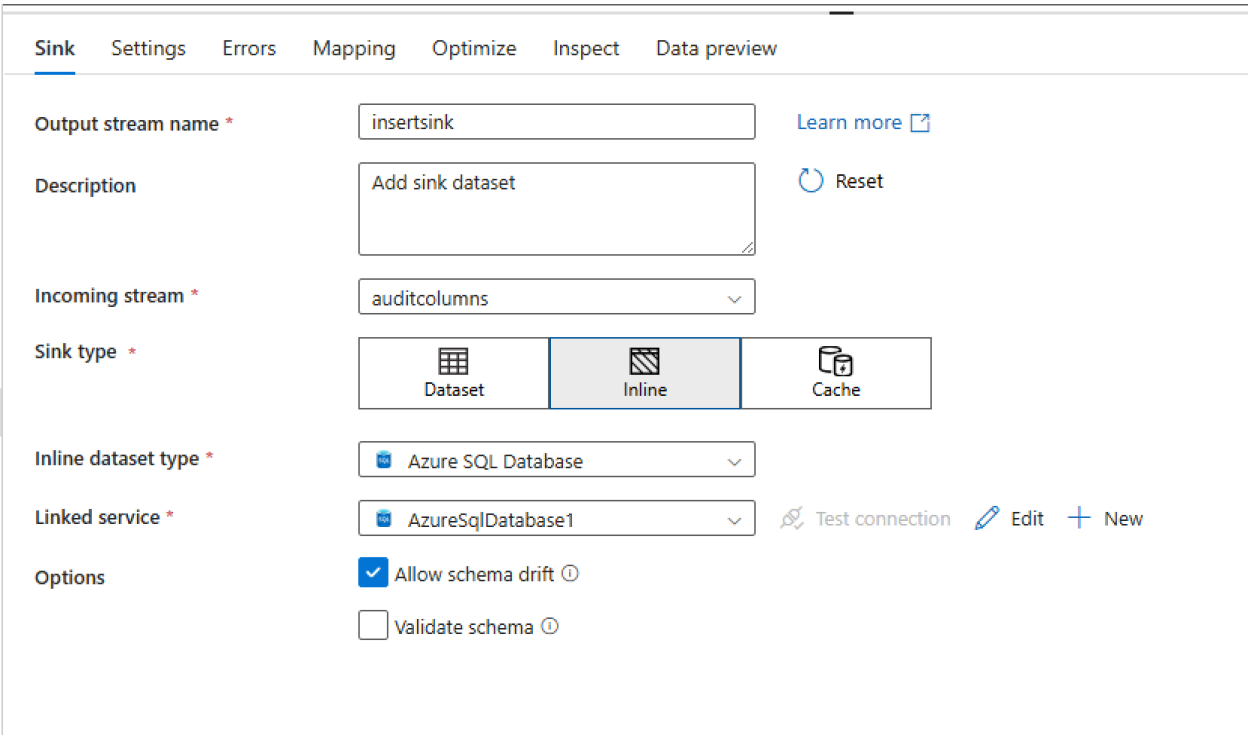
For update we are giving: src_account_id==account_id && src_hashkey!=hashkey



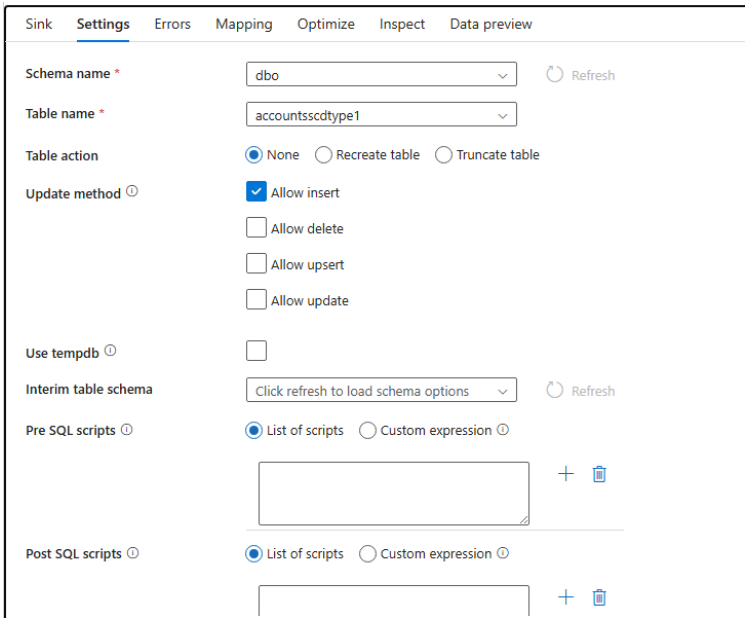
Now we are seeing the derived columns functions on insert side



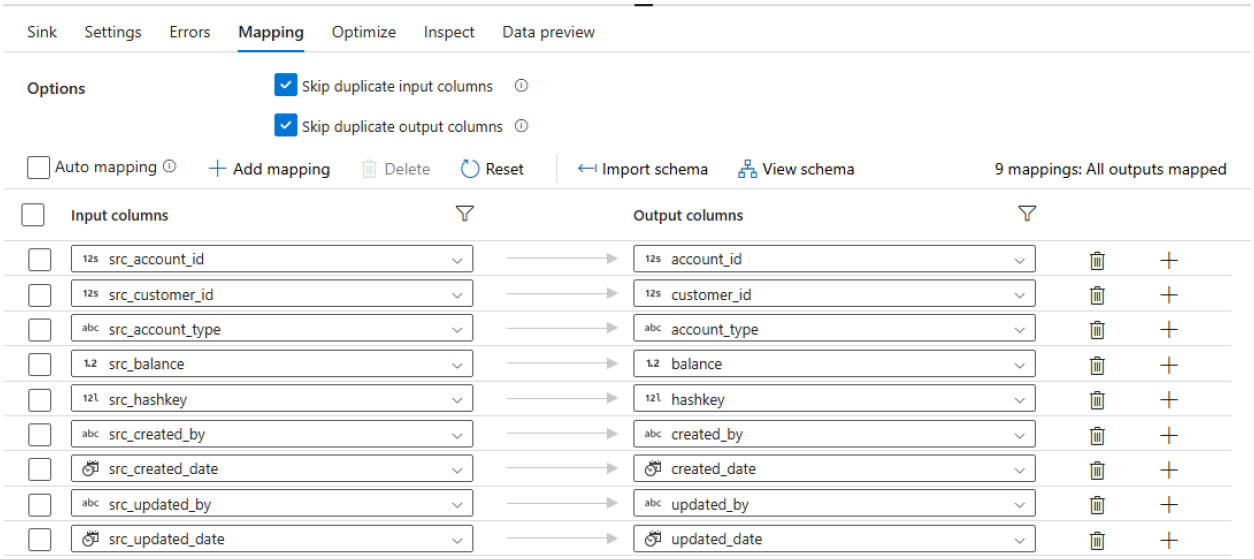
This is derived columns on the update side



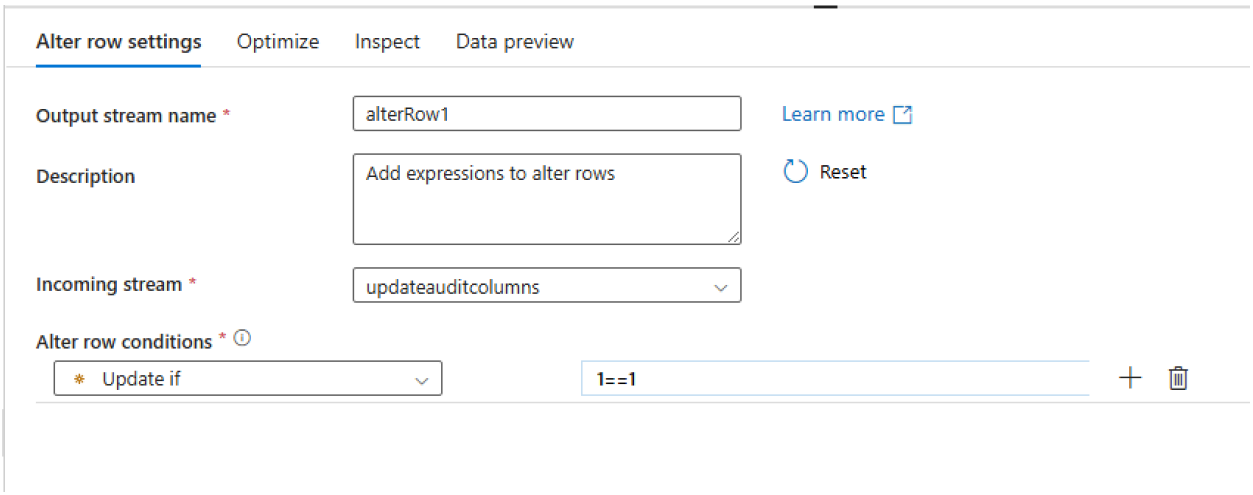
Here is the sink settings we are going to save the scd type 1 transformed data to sql database.



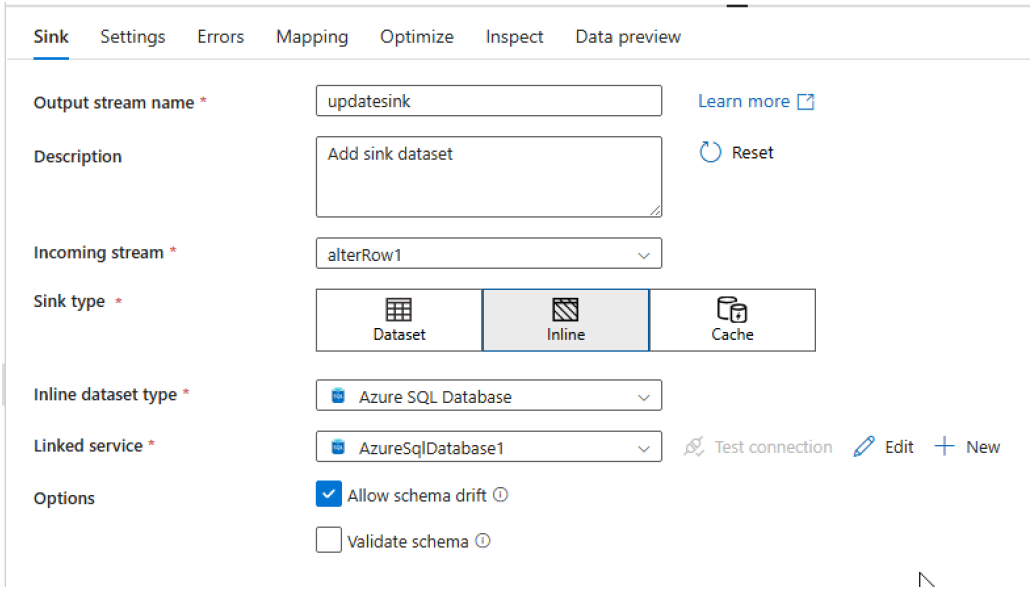
Select the dbo schema and select the table required and click on allow insert for insertion

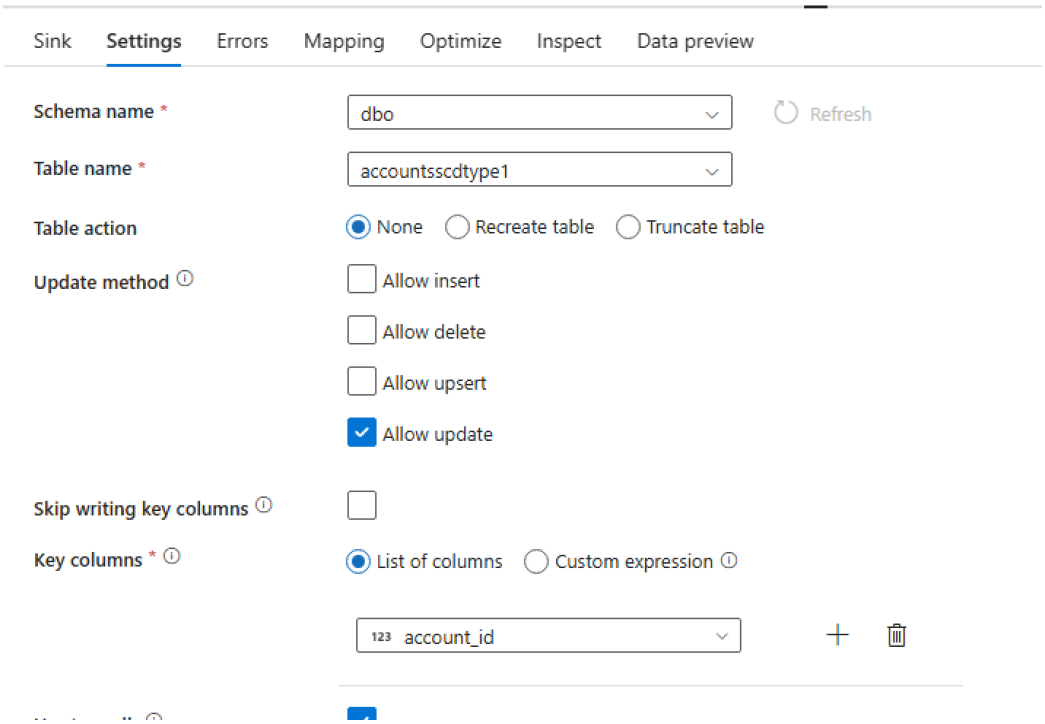


And then set the mapping for the sql table to allow data to be inserted properly

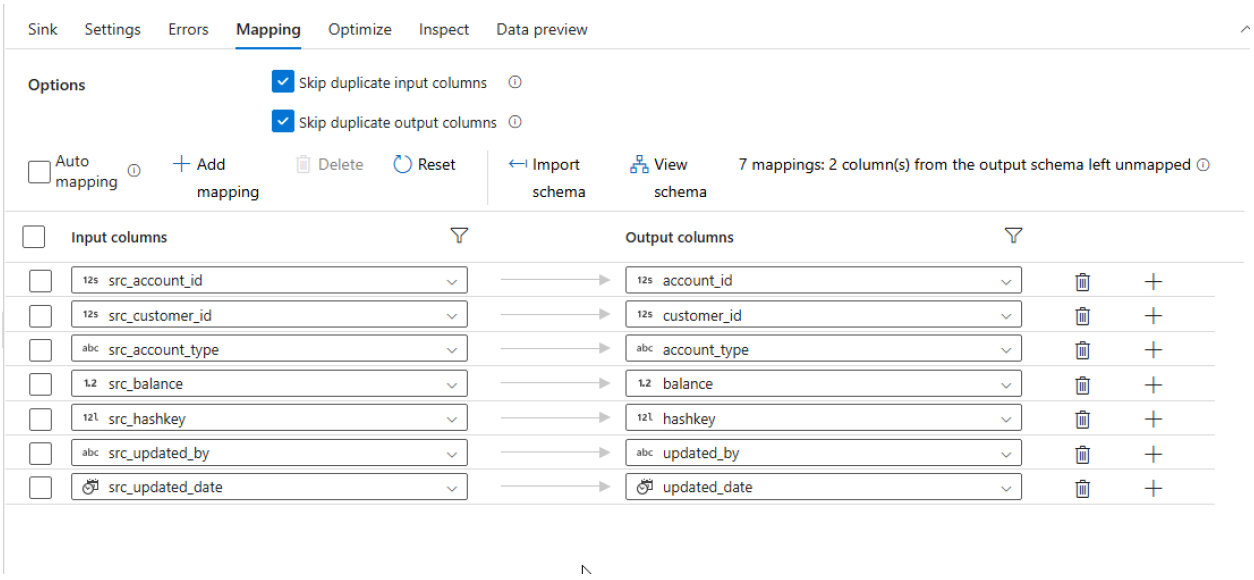


Update condition for updates in scd type 1

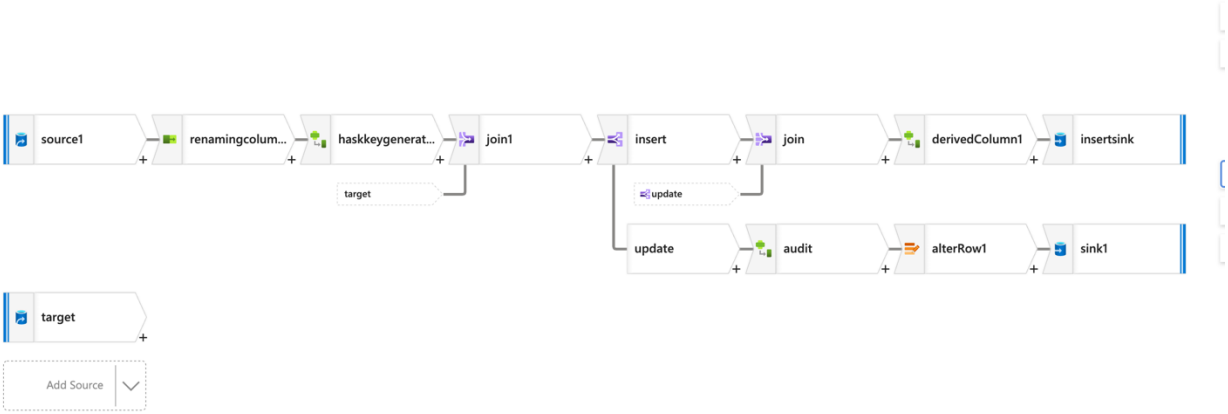




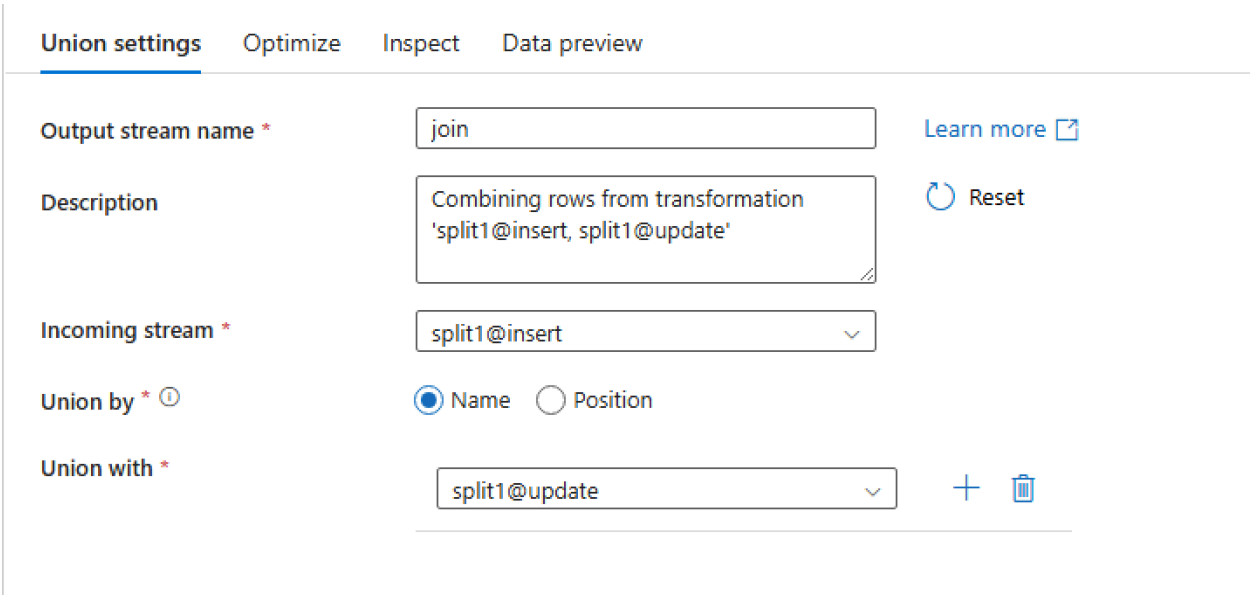
Here is the update side sink details

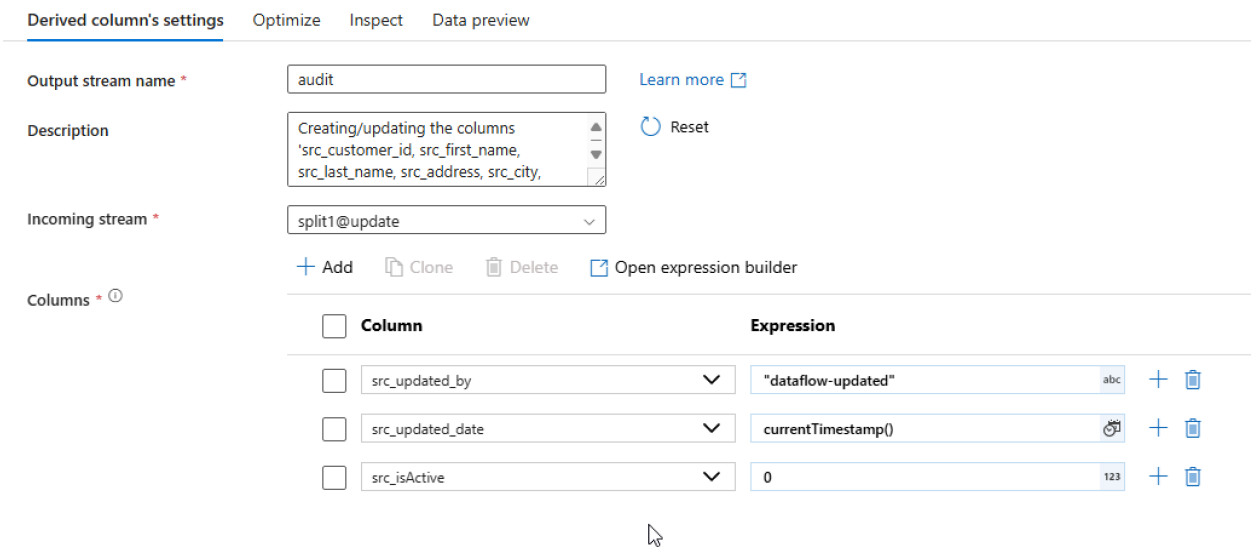


Update side mapping

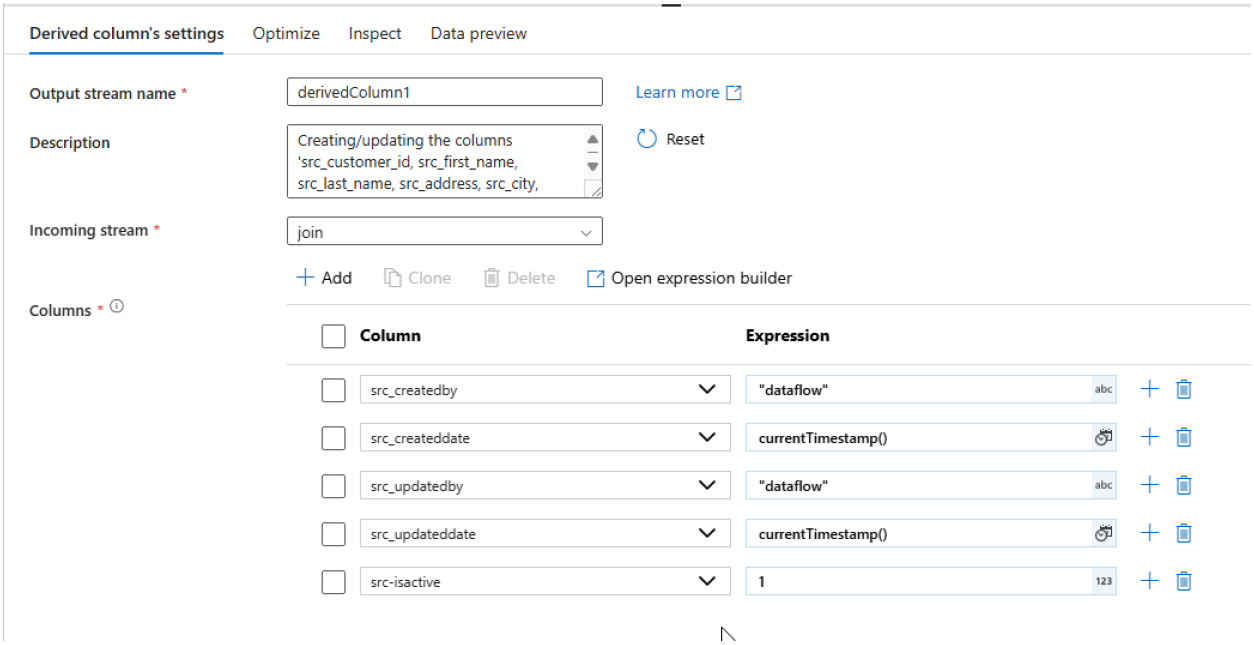


SCD type 2 transformation

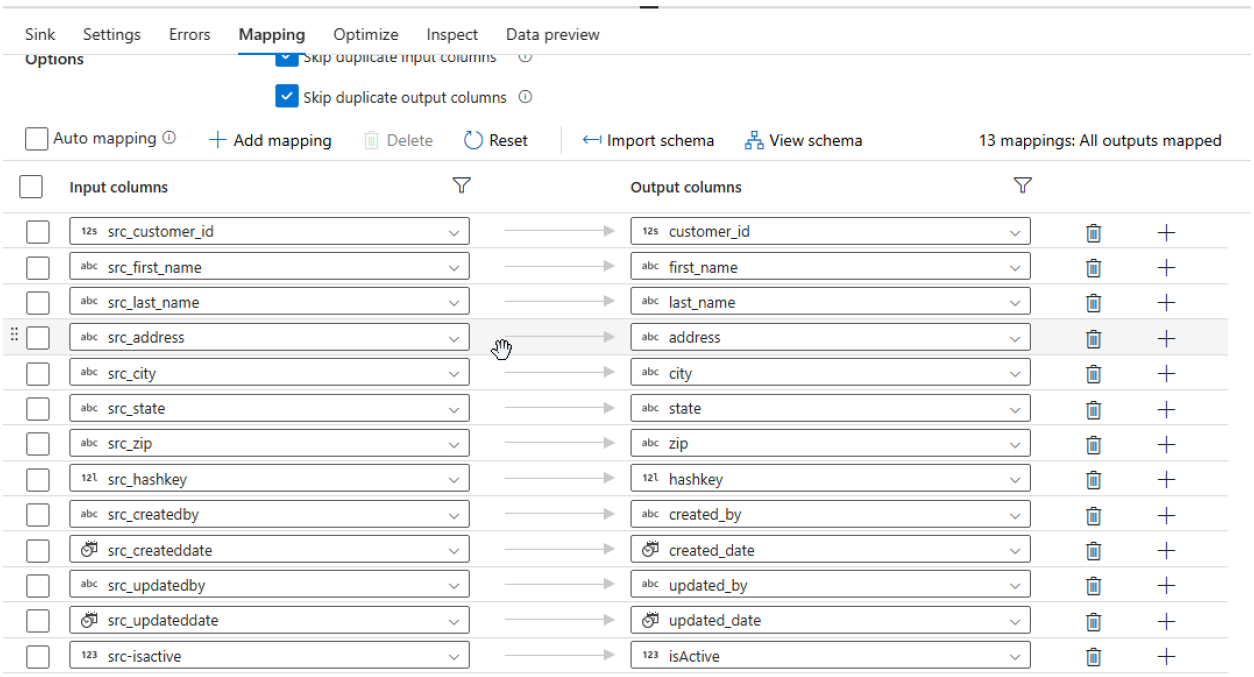




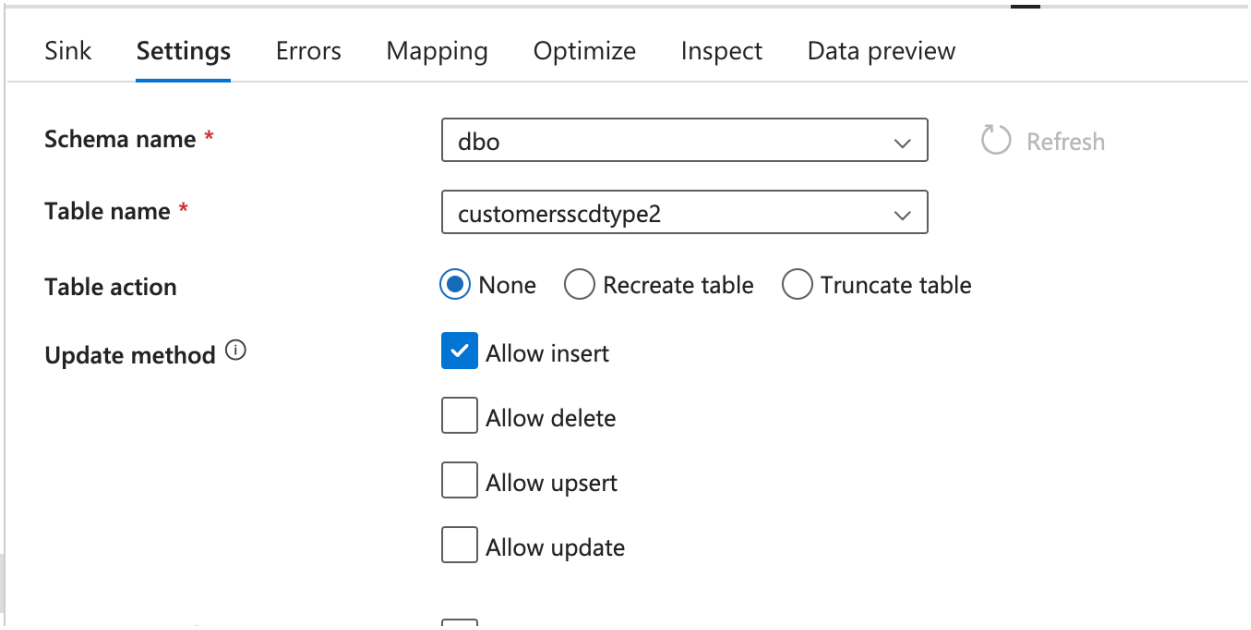
Here is the update side derived column



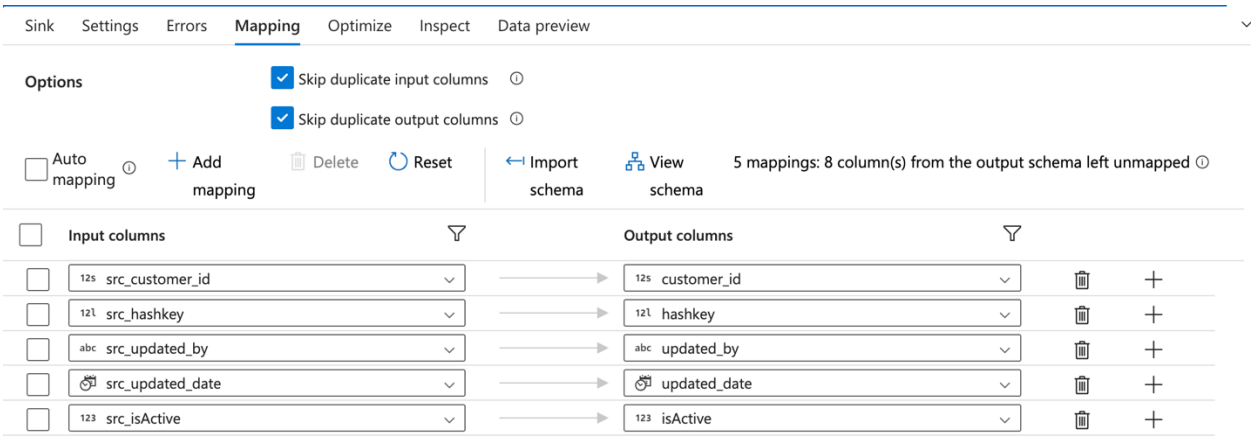
This is the insert side derived columns currenttimestamp() gives todays date and time at the time of insertion



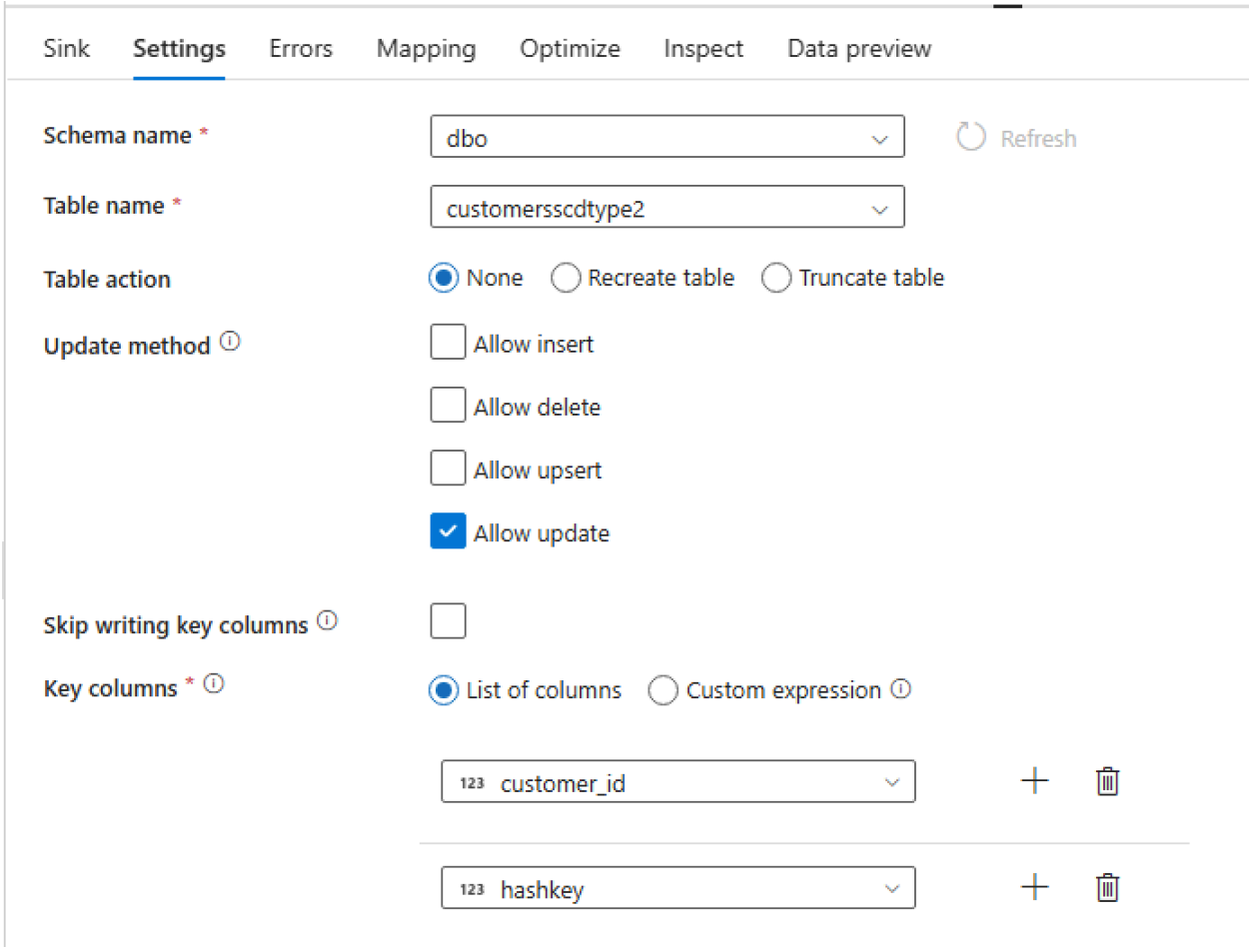
This is the mapping on the insert side of the pipeline



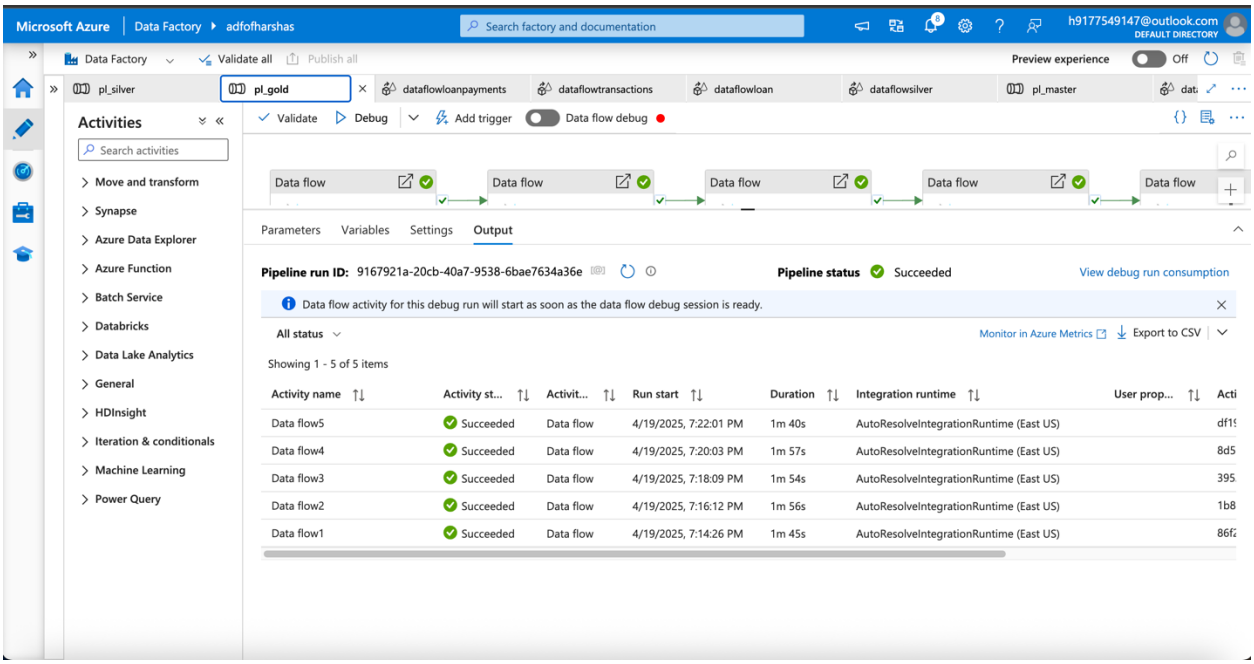
We are allowing the insert function for the insert side of the pipeline



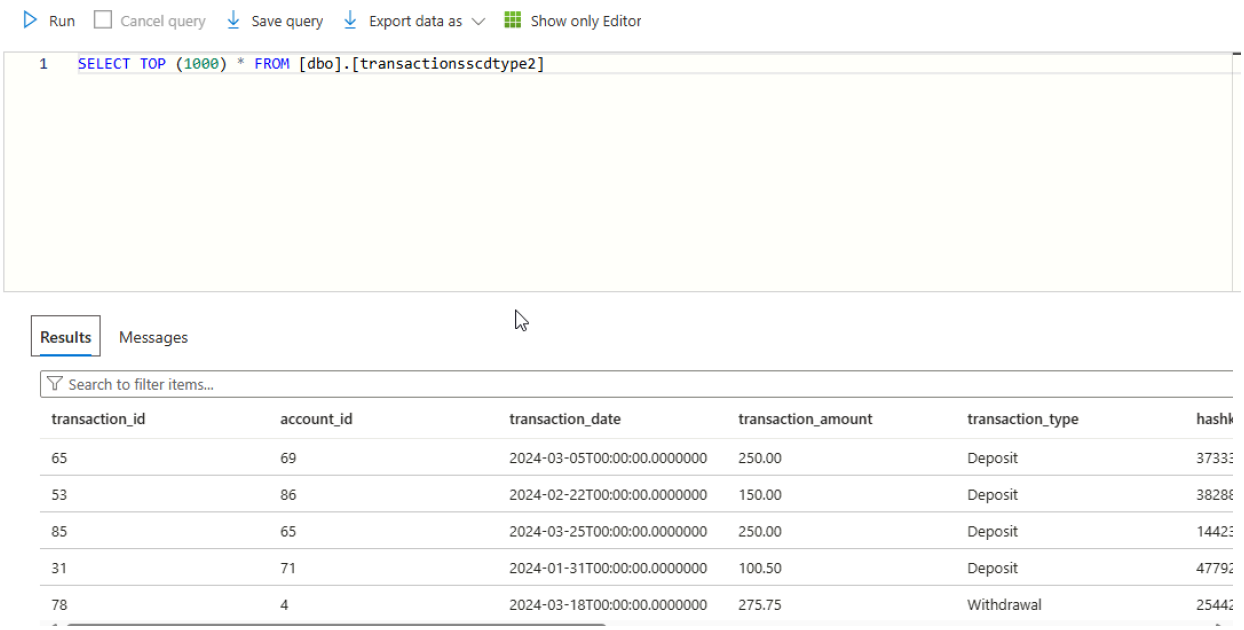
We only update haskey updated_by updated_date and isactive fields only in the update side



We are allowing update and are giving customer_id and hashkey for the key columns



Pipelines successfully run



SCD type 2 data stored successfully in SQL database

Query 1 | Query 4

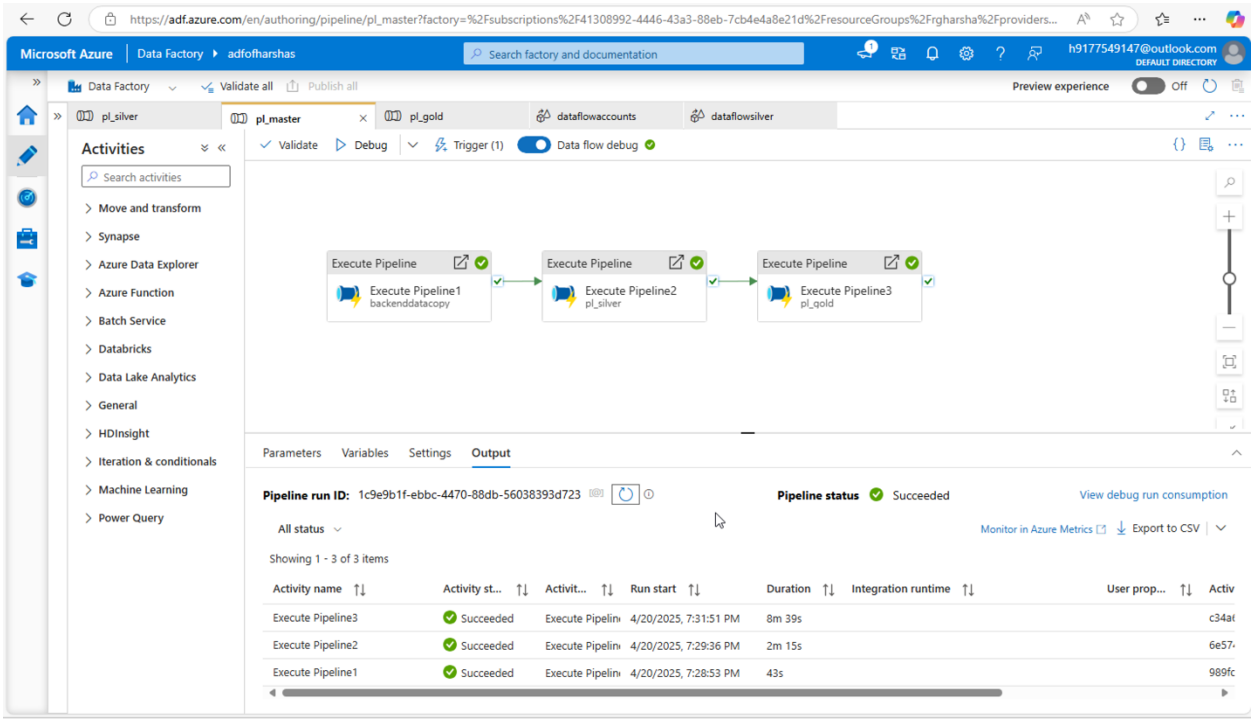
Run | Cancel query | Save query | Export data as | Show only Editor

```
1  SELECT TOP (1000) * FROM [dbo].[accountsscdtype1]
```

Results | Messages

hashkey	created_by	created_date	updated_by	updated_date
2662522274	dataflow	2025-04-20T02:12:37.6510000	dataflow-updated	2025-04-21T02:32:33.1570000
2464207871	dataflow	2025-04-20T02:12:37.6510000	dataflow-updated	2025-04-21T05:04:01.8550000
1122301726	dataflow	2025-04-20T02:12:37.6510000	dataflow-updated	2025-04-21T02:32:33.1570000
630820887	dataflow	2025-04-20T02:12:37.6510000	dataflow-updated	2025-04-21T02:32:33.1570000

SCD type 1 data successfully stored in SQL database.



Triggers

To execute a pipeline set the trigger. Triggers represent a unit of processing that determines when a pipeline execution needs to be kicked off.

+ New Refresh

Filter by name Annotations : **Any**

Showing 1 - 1 of 1 items

Name	Type	Status	Related	Annotations
pl_master_trigger	Schedule	Started	1	