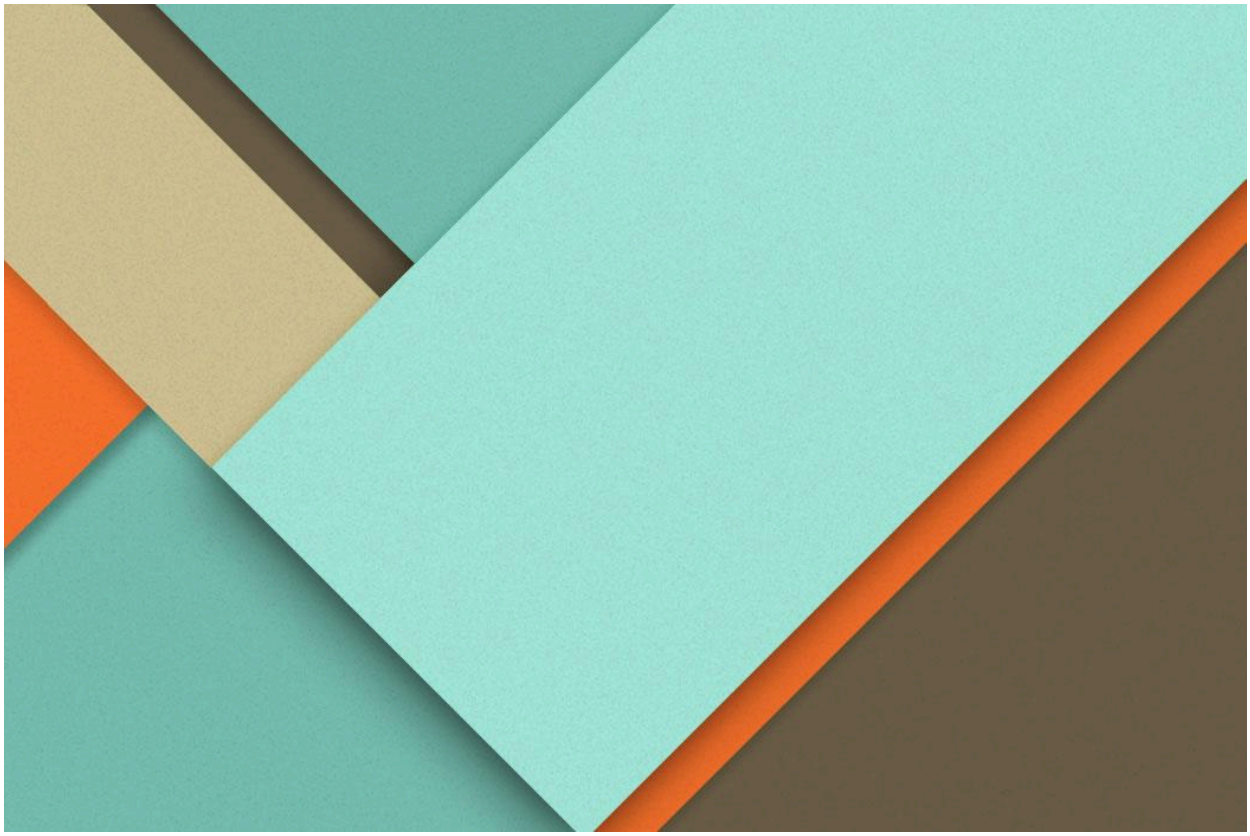


Project Name: Data Pipeline for Customer Account Analysis.

—

Sudhanshu Kharbanda

Objective

The objective of this project is to design and implement a robust and scalable **data pipeline** for processing **customer account data**. The pipeline performs the following functions:

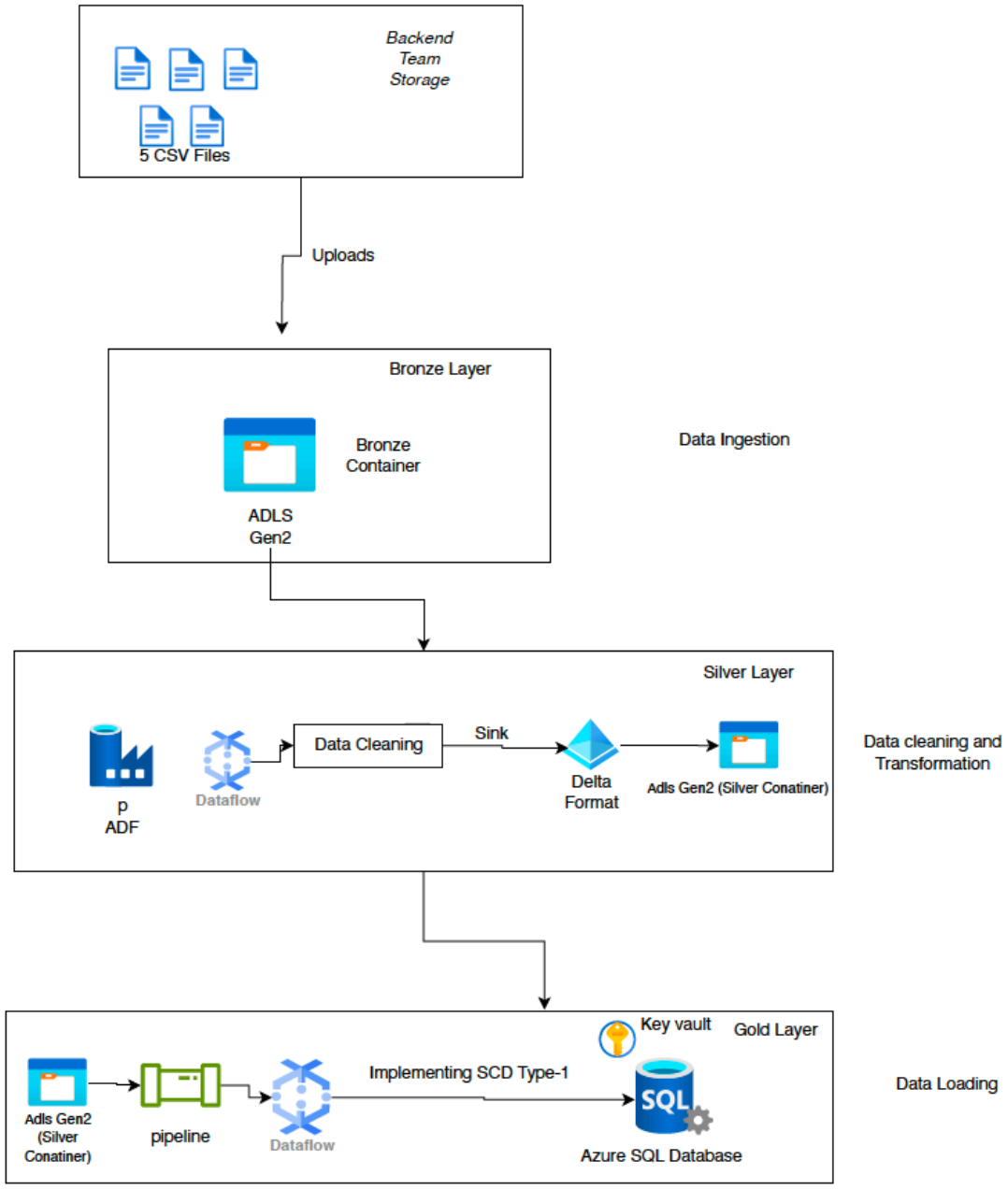
- **Data Ingestion:** Copy raw data from a backend team's Azure Storage Account to a **Data Lake Raw (Bronze) container**.
- **Data Cleaning & Transformation:** Use **Azure Data Factory (ADF)** Data Flows to clean and transform the ingested data.
- **ETL Processing:** Prepare curated (Silver) datasets and upsert (Insert or Update) them into a **SQL Database** using **Slowly Changing Dimension Type 1 (SCD-1)** logic.
- **Security:** Secure credentials and connection strings using **Azure Key Vault**.

This solution ensures **accuracy, efficiency, and reliability** in data handling, enabling downstream analytics and reporting.

Tools & Services Used

1. **Azure Data Factory (ADF)** – For data ingestion, cleaning, transformation, and loading.
2. **Azure Data Lake Storage (ADLS)** – Organized in a **Bronze–Silver–Gold** architecture.
3. **Azure SQL Database** – Final destination for cleaned and transformed data.
4. **Azure Key Vault** – Secure storage for credentials and connection strings.
5. **Delta/Parquet File Format** – Optimized storage format for transformations and analytics.
6. **Draw.io**- For creating an Architecture diagram.

Specifications Project Workflow and Architecture diagram



Step 1 – Data Ingestion (Backend Storage → Bronze Layer)

- **Purpose:** Copy raw data from the backend team's Azure Storage account to the Bronze container in ADLS.
- **Process:**
    - Used **ADF Copy Activity** to transfer files from the backend team's container to the Bronze layer.
    - Source Files:
        - `accounts.csv`
        - `customers.csv`
        - `loan_payments.csv`
        - `loans.csv`
        - `transactions.csv`
    - Sink: ADLS Raw (Bronze) Container.

Step 2 – Data Cleaning & Transformation (Bronze → Silver Layer)

- **Purpose:** Ensure data quality before further processing.
- **Actions Taken:**
    - Read data from Bronze container.
    - Applied transformations using **ADF Data Flow**:
        - Removed duplicate rows.
        - Null checks for primary keys
    - Stored cleaned datasets in the **Curated (Silver) container** in **Delta format**.

Step 3 – Loading into SQL DB with SCD-1 Logic (Silver → SQL DB(Gold))

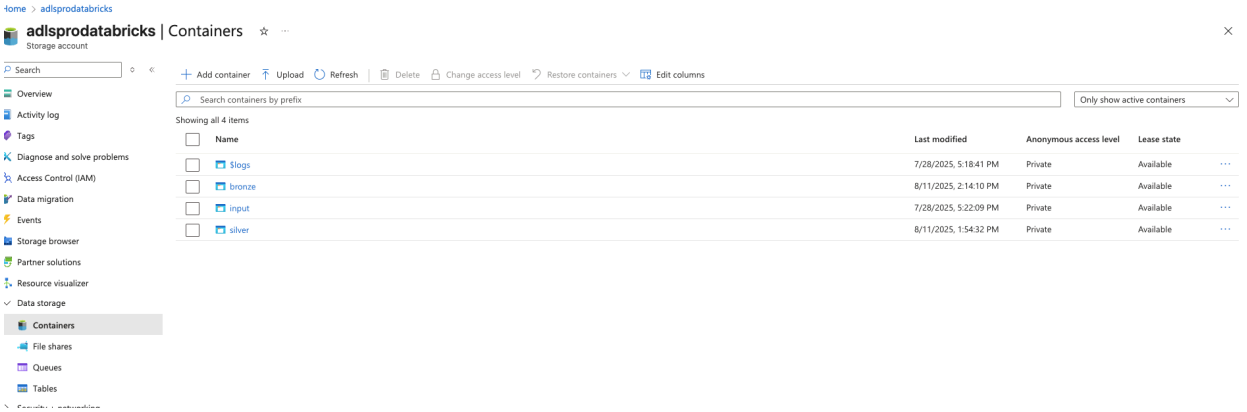
- **Purpose:** Maintain updated records by replacing old data with new updates.
- **Process:**
    - Created an **ADF Pipeline** to read Delta files from the **Silver** layer.
    - Implemented **SCD Type 1**:
        - **Insert** new records if they do not exist in the target table.
        - **Update** existing records if primary key matches.
    - Executed pipeline to load initial dataset into SQL DB.
    - Modified source file (Bronze) to:
        - Add a **new record**.
        - Update an **existing record**.
    - Reran the pipeline to verify **upsert functionality** worked as expected.

Step 4 - Security Implementation

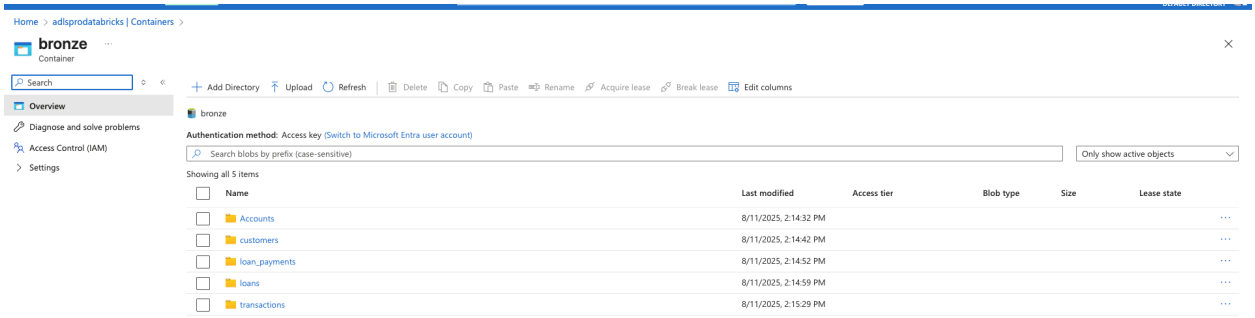
- All connection strings, SQL credentials, and storage keys were securely stored in **Azure Key Vault**.
- ADF pipelines referenced secrets dynamically from Key Vault, preventing hardcoding of sensitive information.

Implementation screenshots

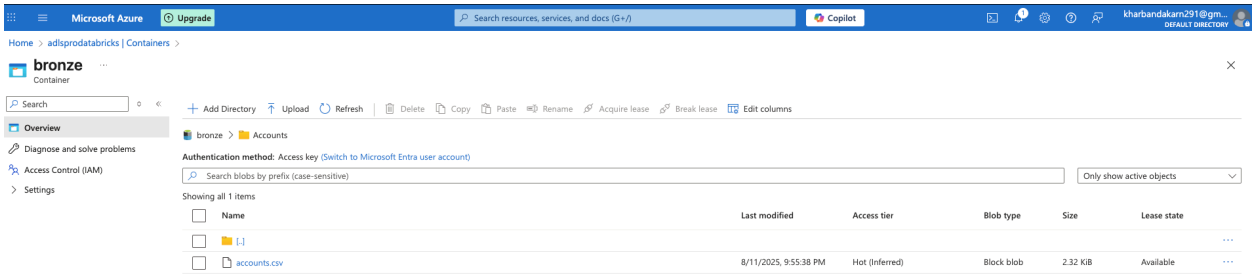
Creating Silver and Bronze containers in the ADLS Gen2



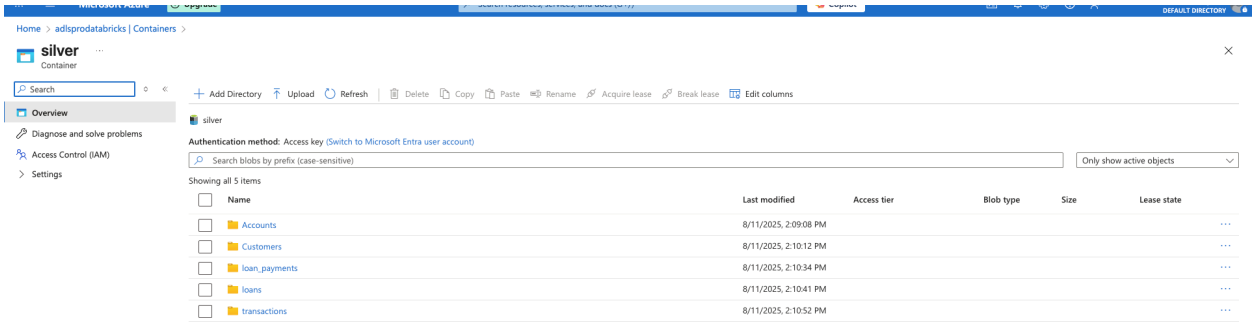
Inside the bronze containers 5 directories has been created with specific directories for each csv file.



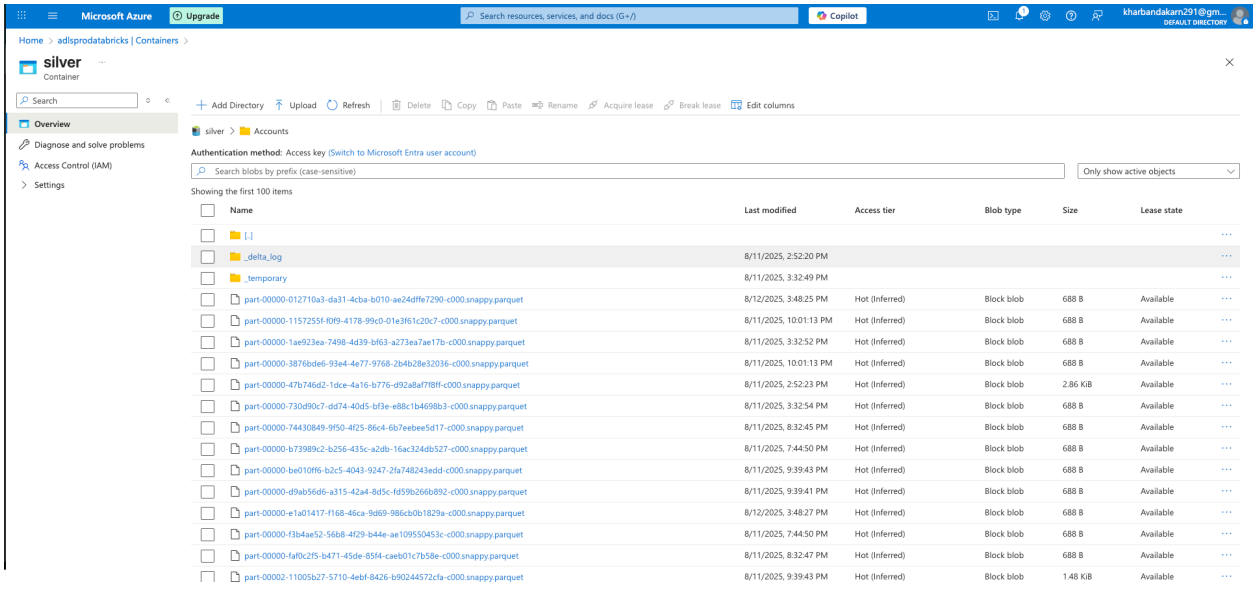
Here each csv files has been uploaded to their specific folders.



Creating a silver container and creating directories for each of the files.

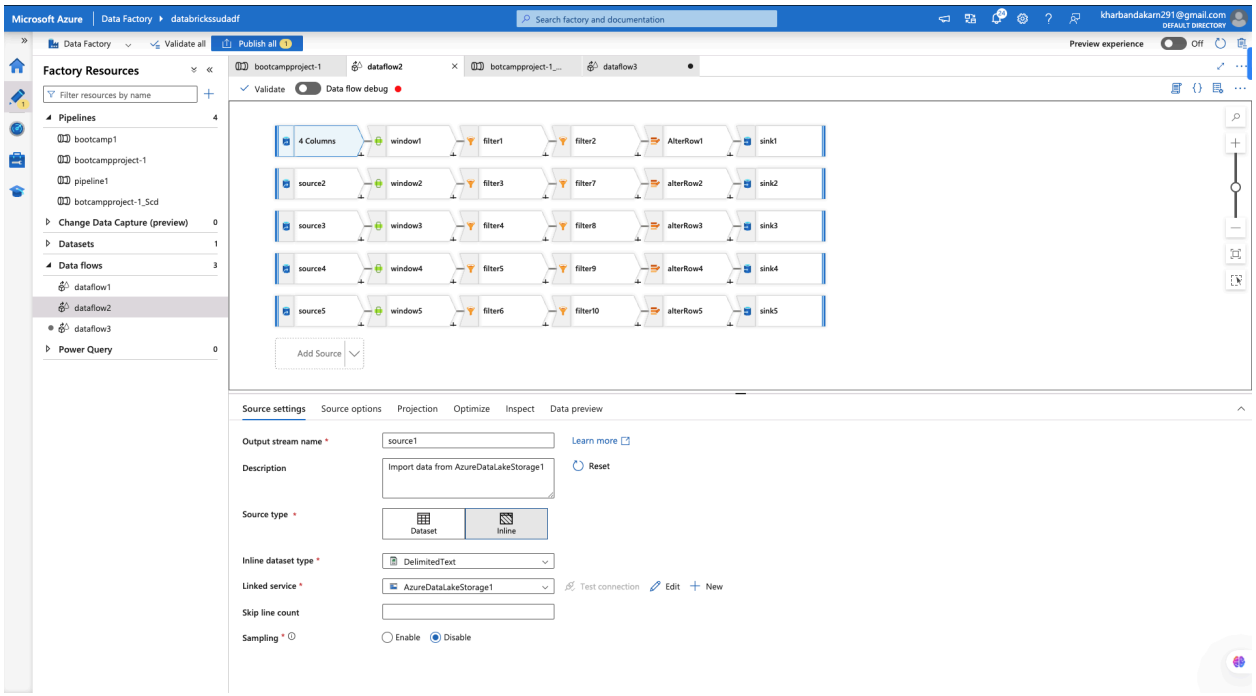


After data cleaning the file will be saved in delta format in the silver container.

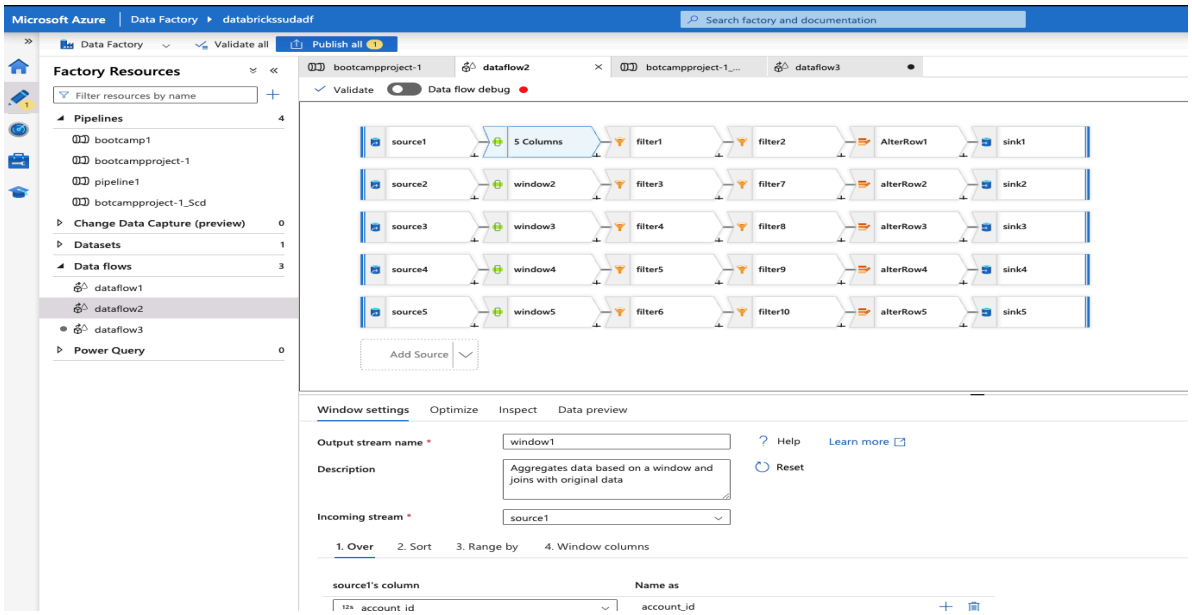


Here are the five pipelines which is created in a single data flow where we are removing duplicates and null values for the primary column of each csv files.

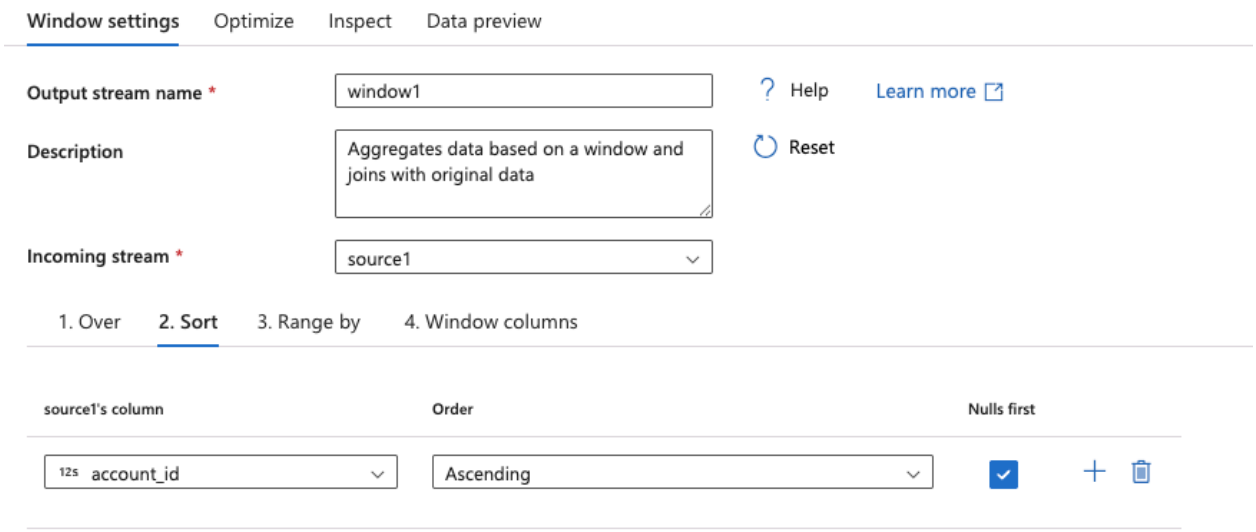
For the source select source dataset type as delimited text and select azure datalake storage as linked service.



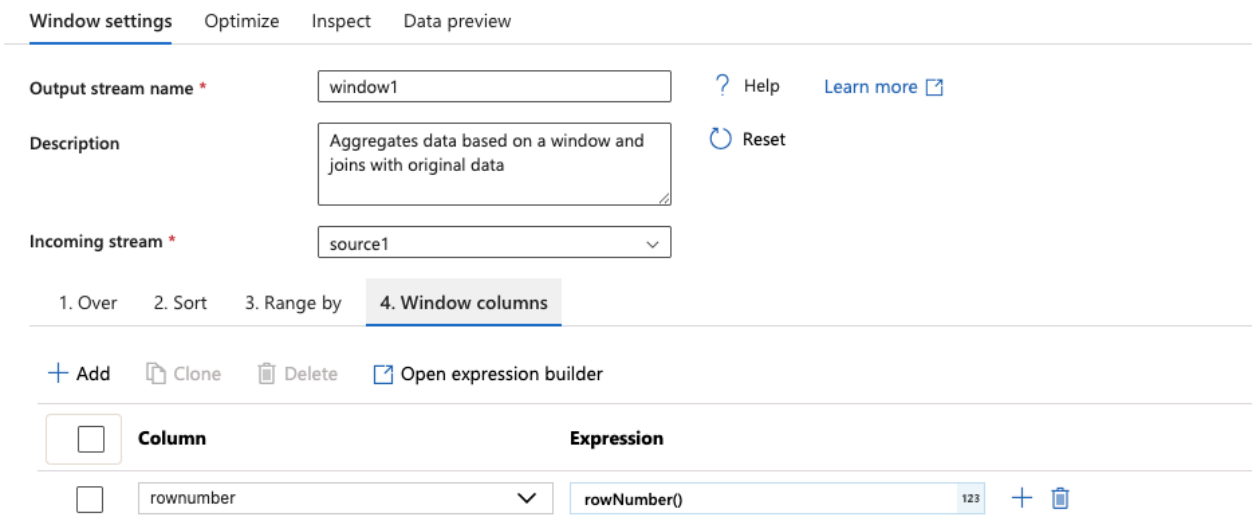
After loading the data window activity is used for removing the duplicates where account id column is choosed where window function is applied.



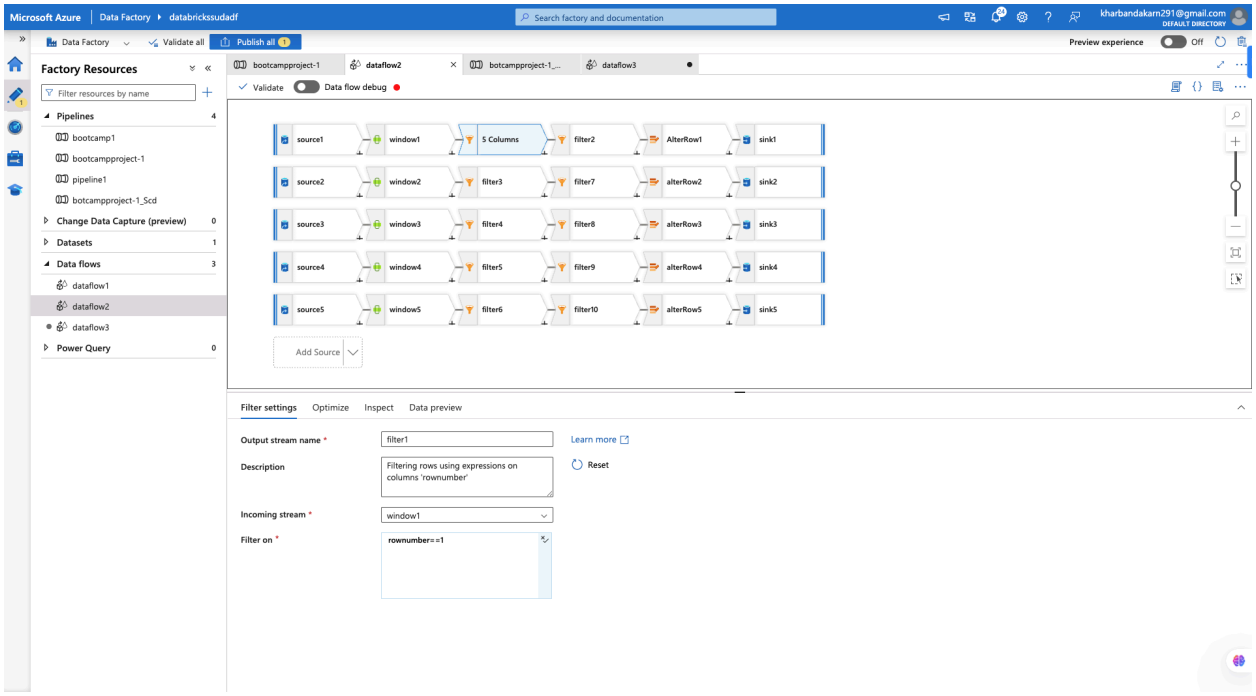
All the partitions are storted according to the accountid in ascending order.



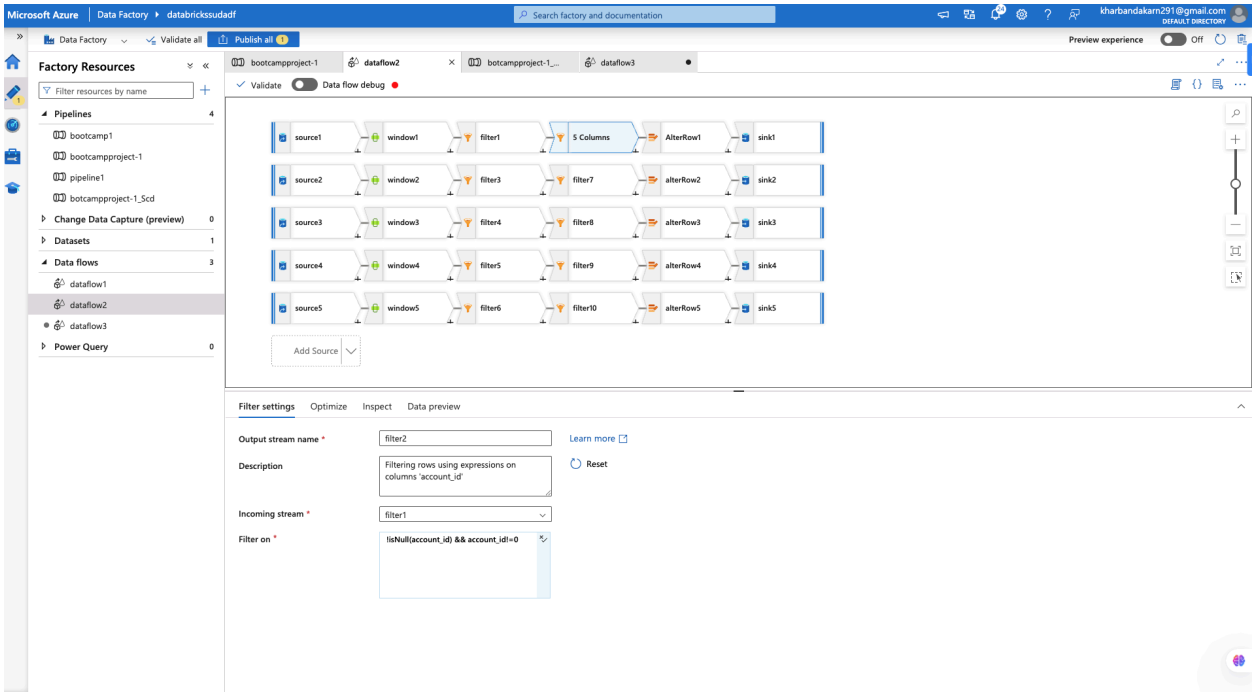
Creating a column which will give rank to eacg element in partition.



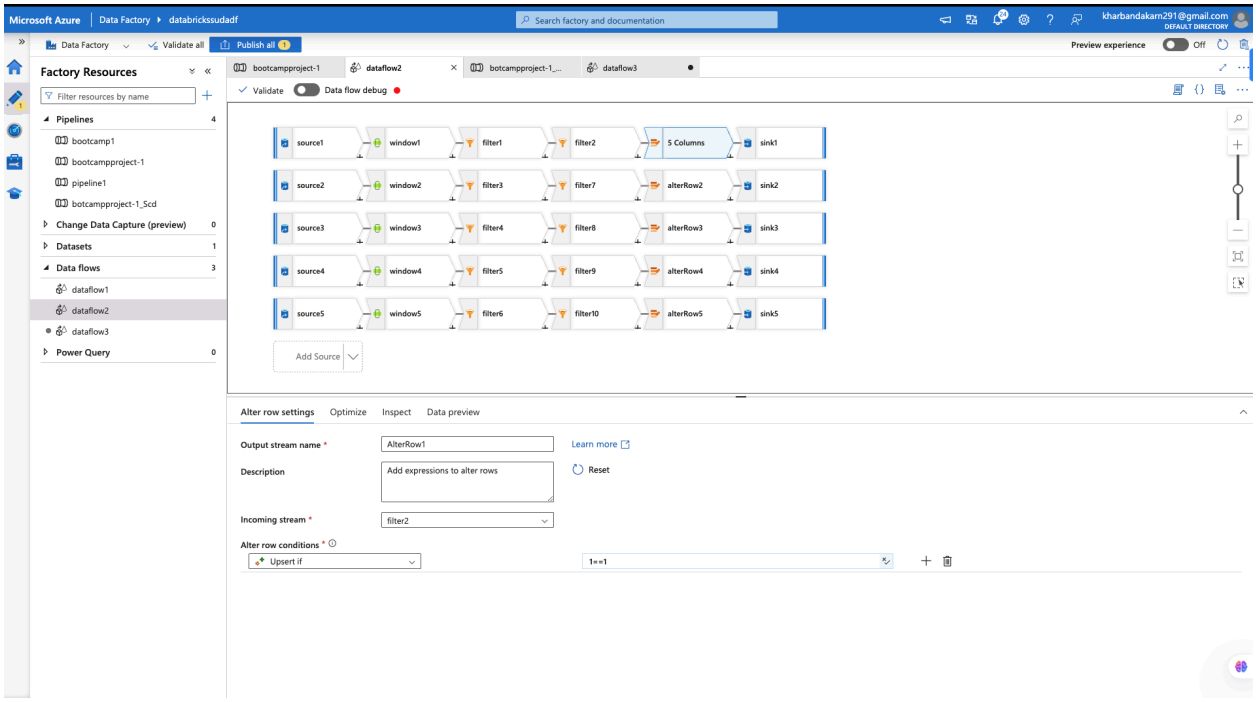
Now filter activity is added where we are filtering rows that is having rownumber=1, basically here we are filtering duplicates rows.



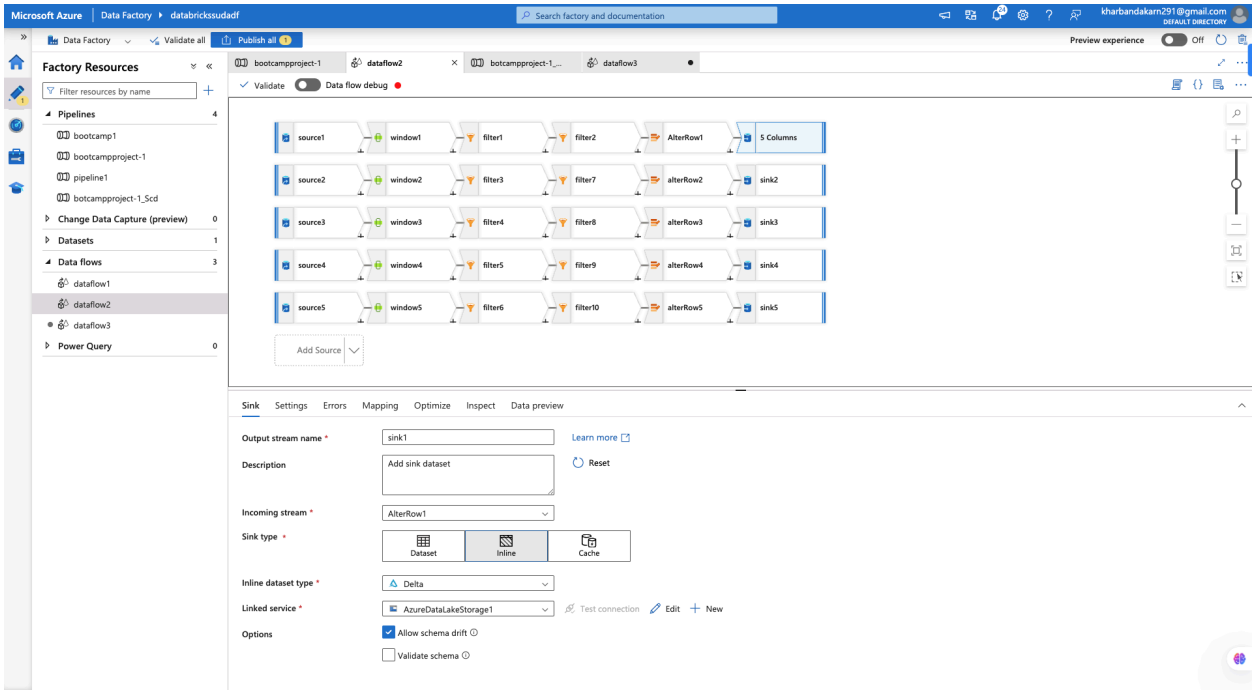
In this filter activity we are removing all the account id with null values and having value = 0.



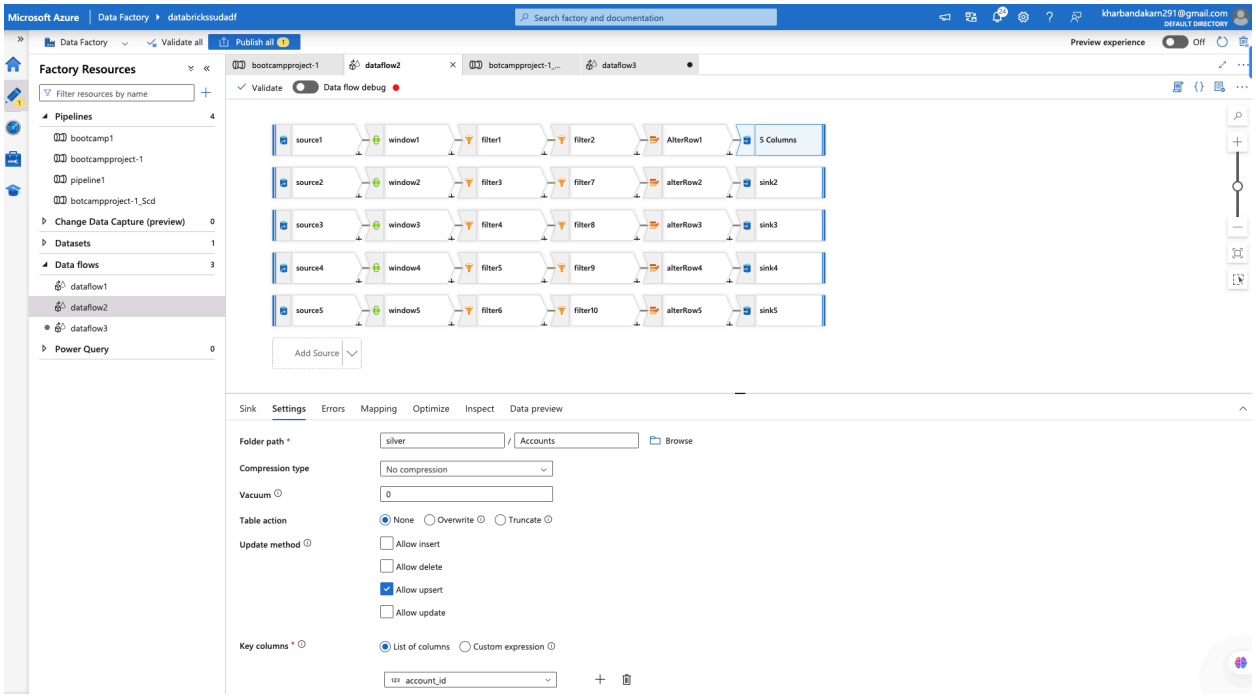
Since we are implementing upsert in sink activity alter row activity is added where we alter row conditions as upsert.



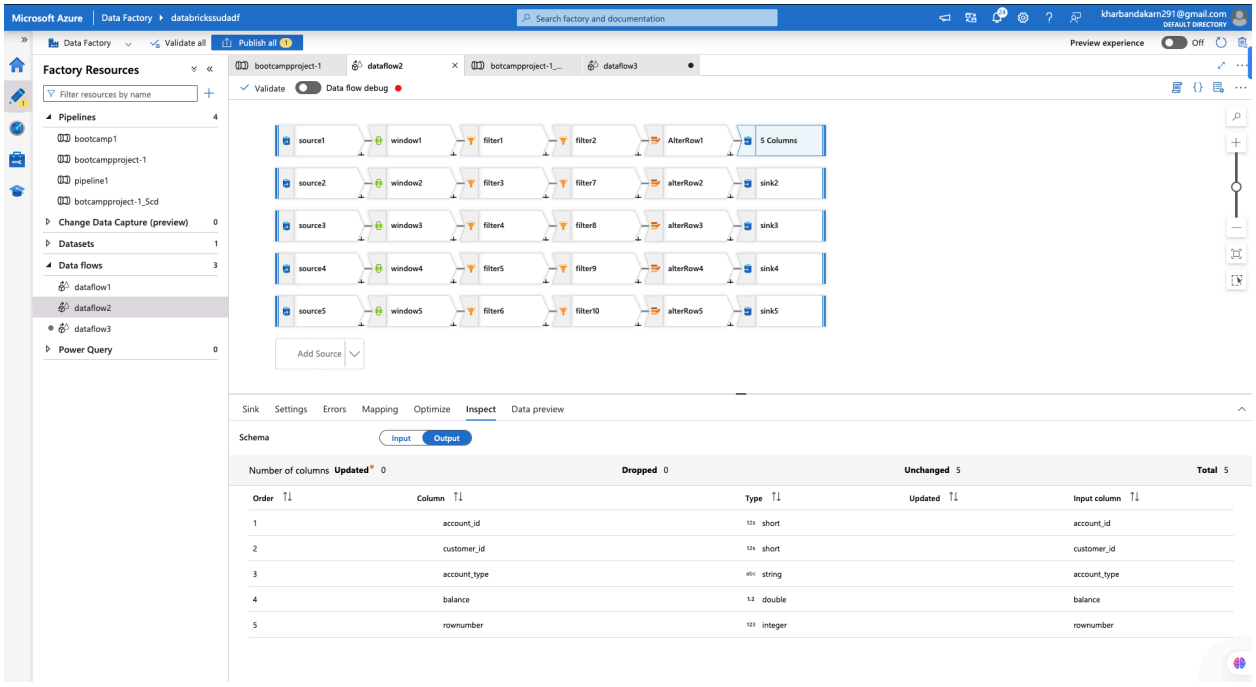
Now we are adding sink activity where specifying the format in which data has been saved.



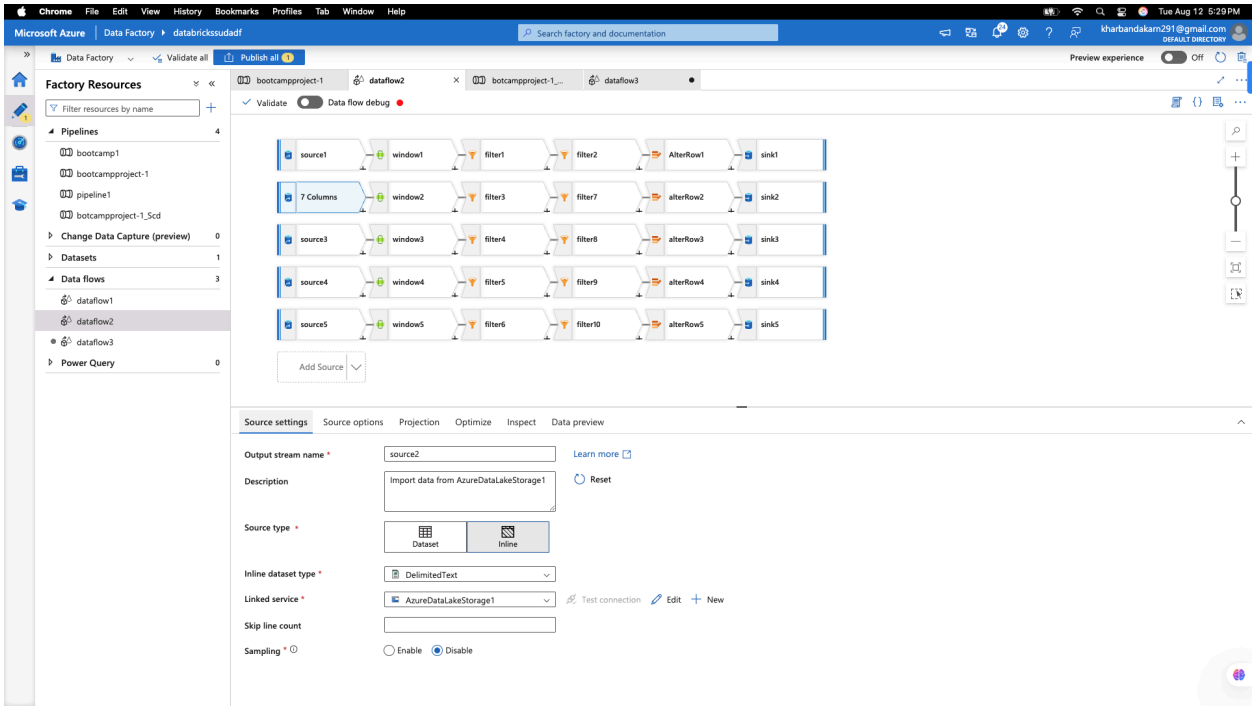
Here we are providing the path of the silver container where all the data has been stored. Here upsert method will be used since there will be updated and new rows will be added into the system.



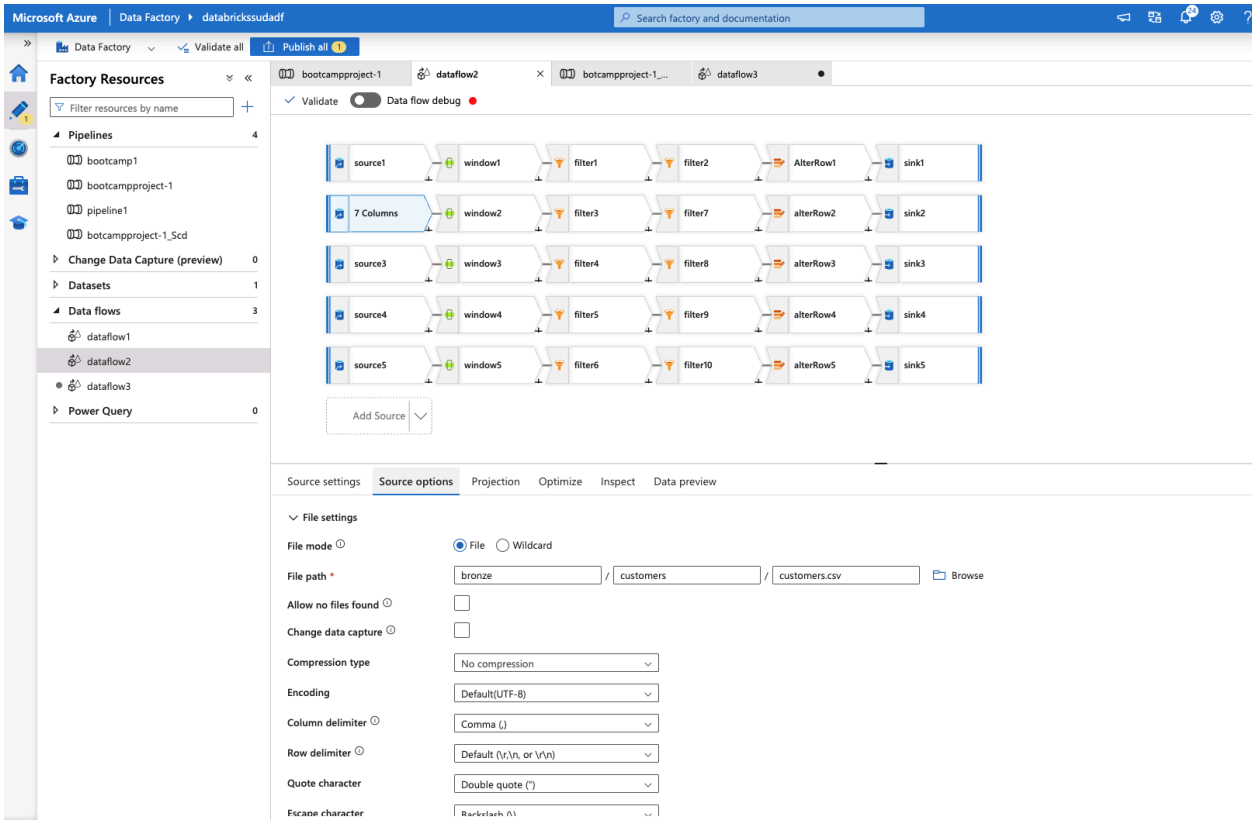
Here structure of the data that has been saved in the delta format.



Now we are adding source activity for loading second file.

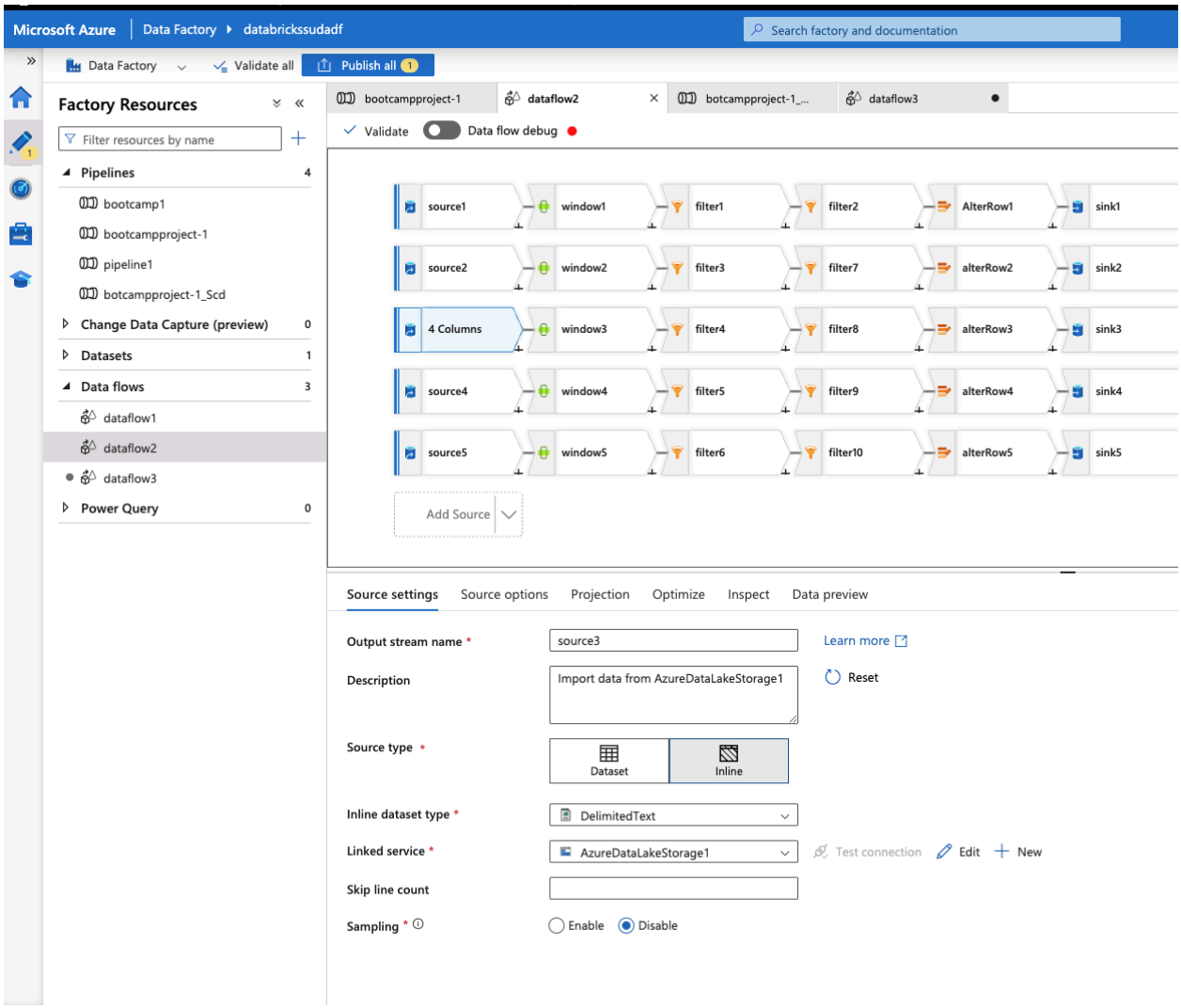


Here we are providing location of the file from where it has to be loaded from bronze layer.

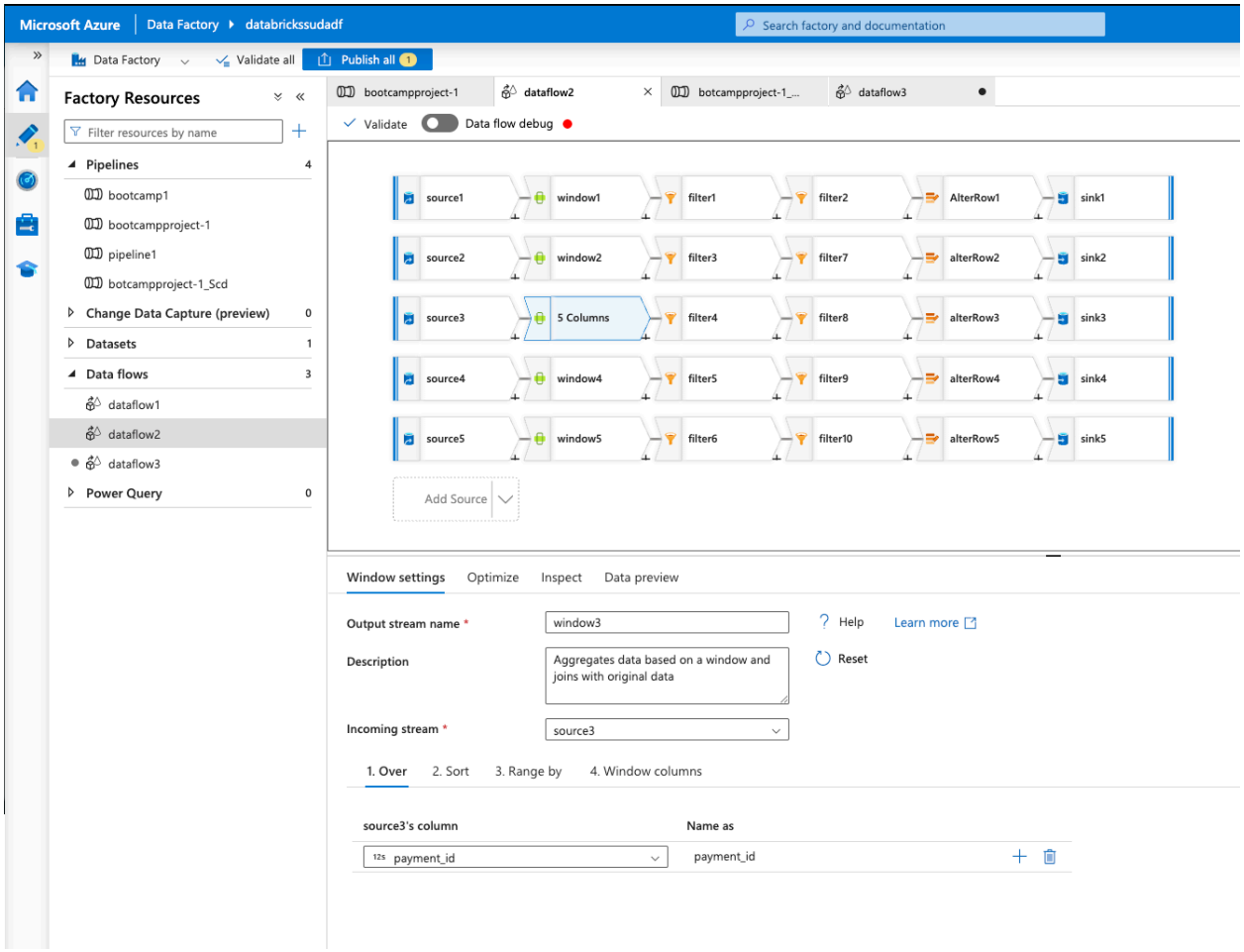


We will be doing filtering of data and saving it to the silver layer in the same way that we have implemented in previous steps. So only source activities for the other pipelines will shown below.

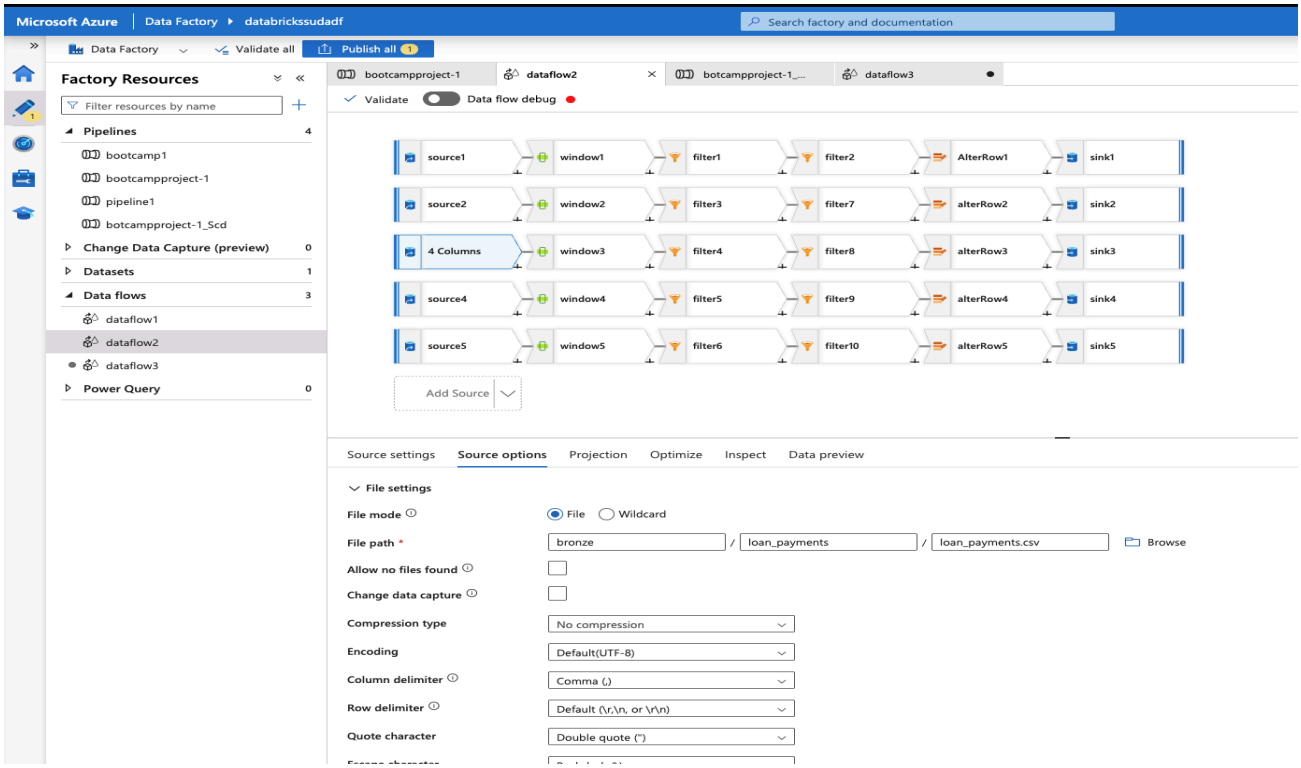
Here is the activity that will be loading the payments.csv file.



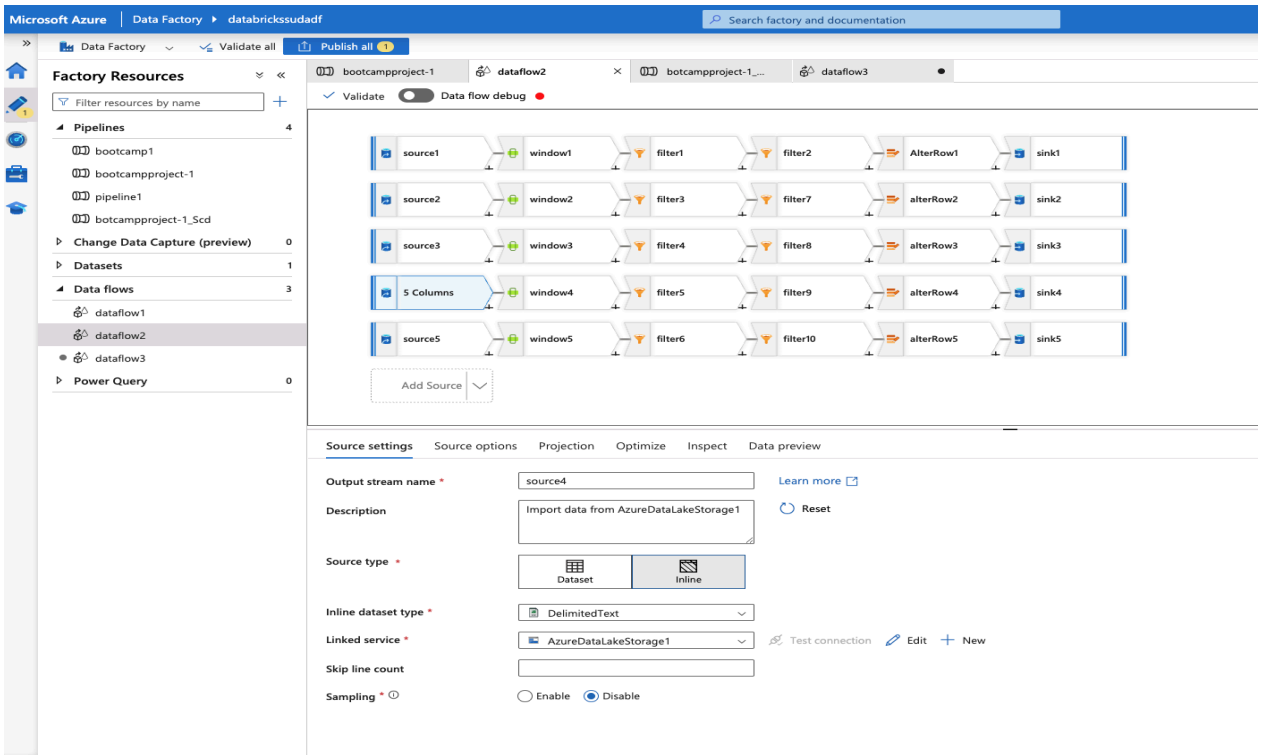
Here windows activity is checking the duplicates rows.



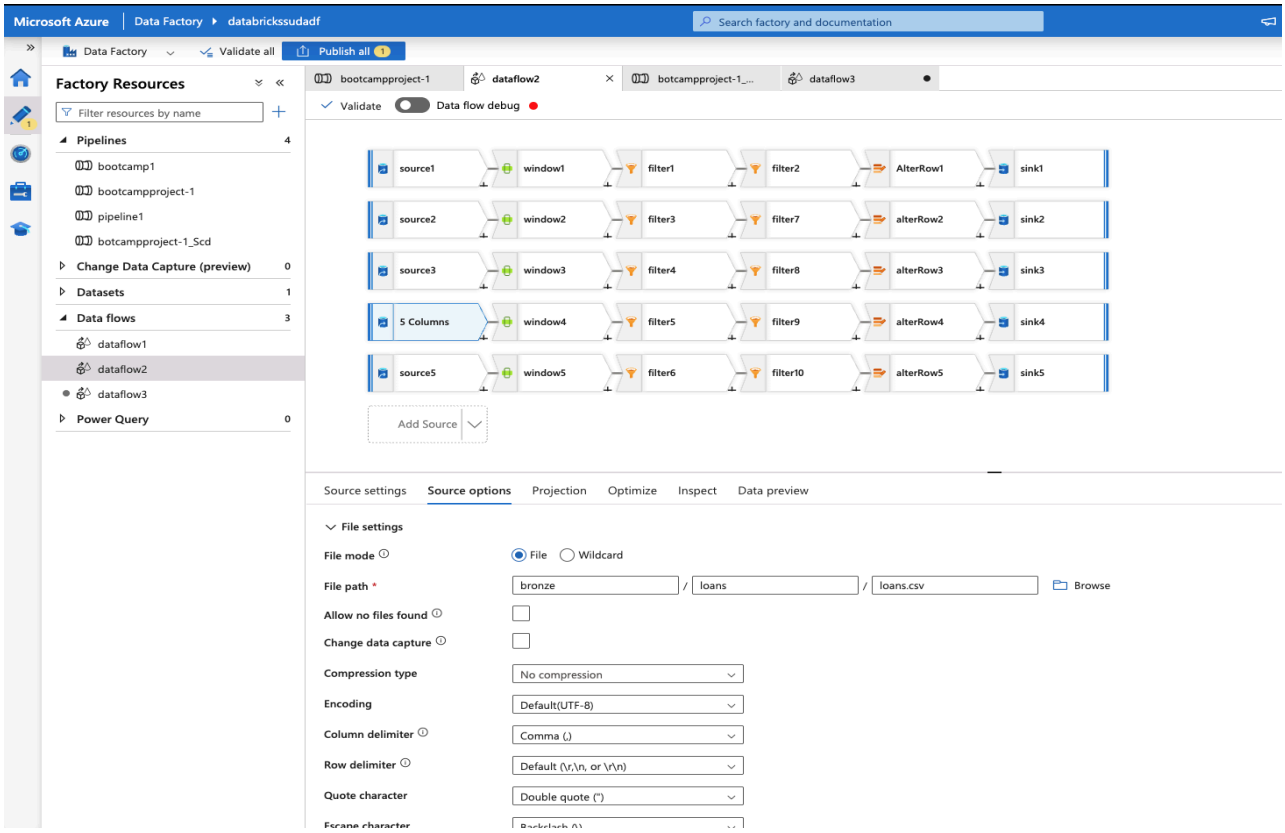
Here is the path from where loan_payments.csv file is loaded.



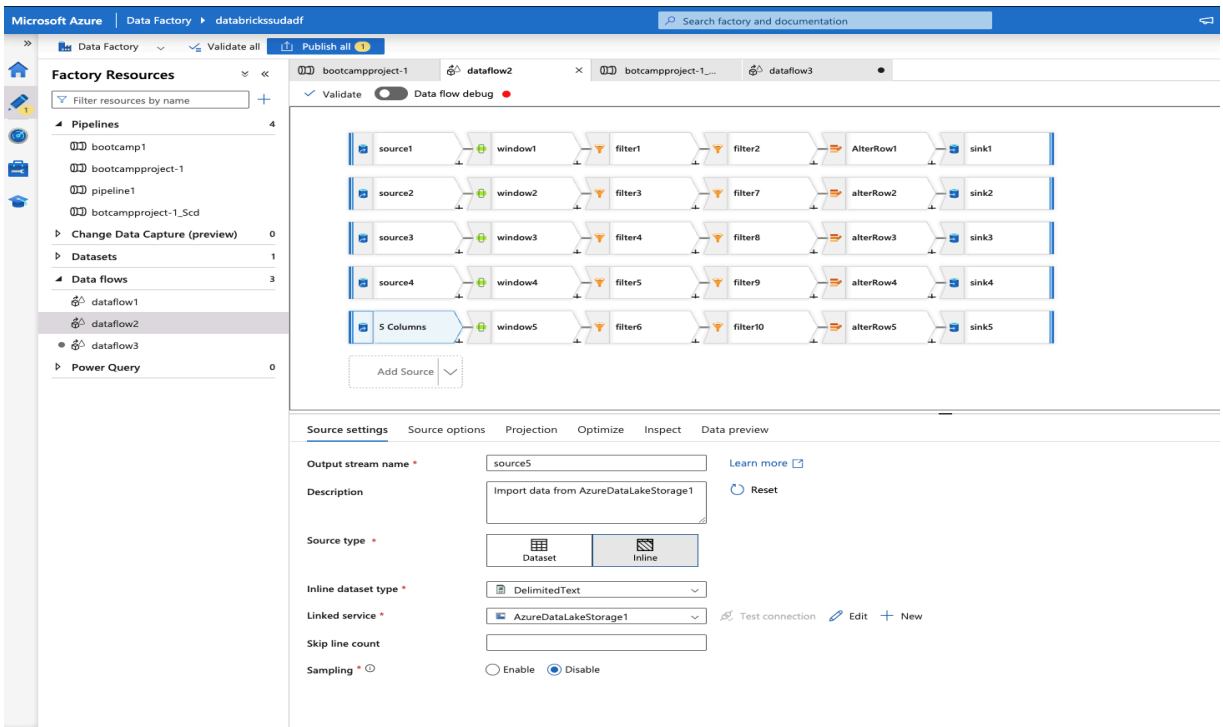
Here we are defining source for Fourth csv file.



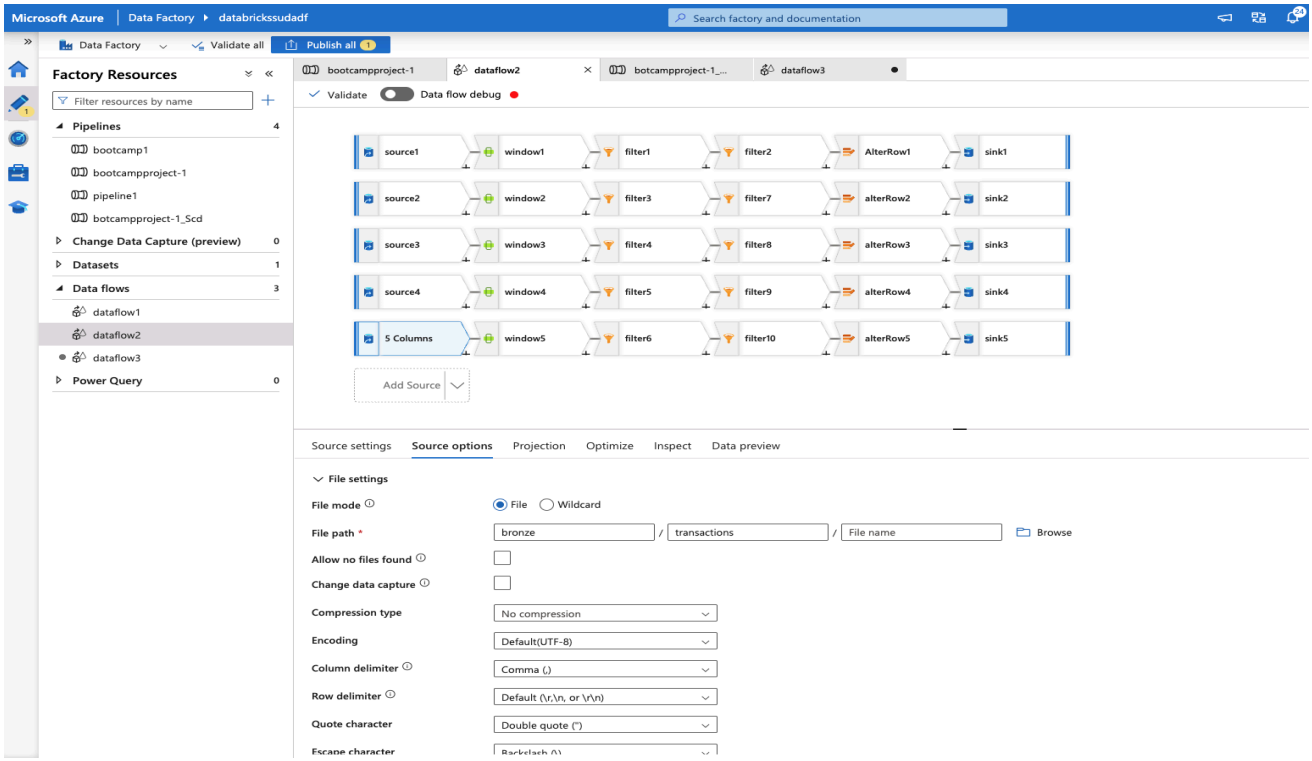
Here we are providing the path for loans.csv file from where it has been loaded.



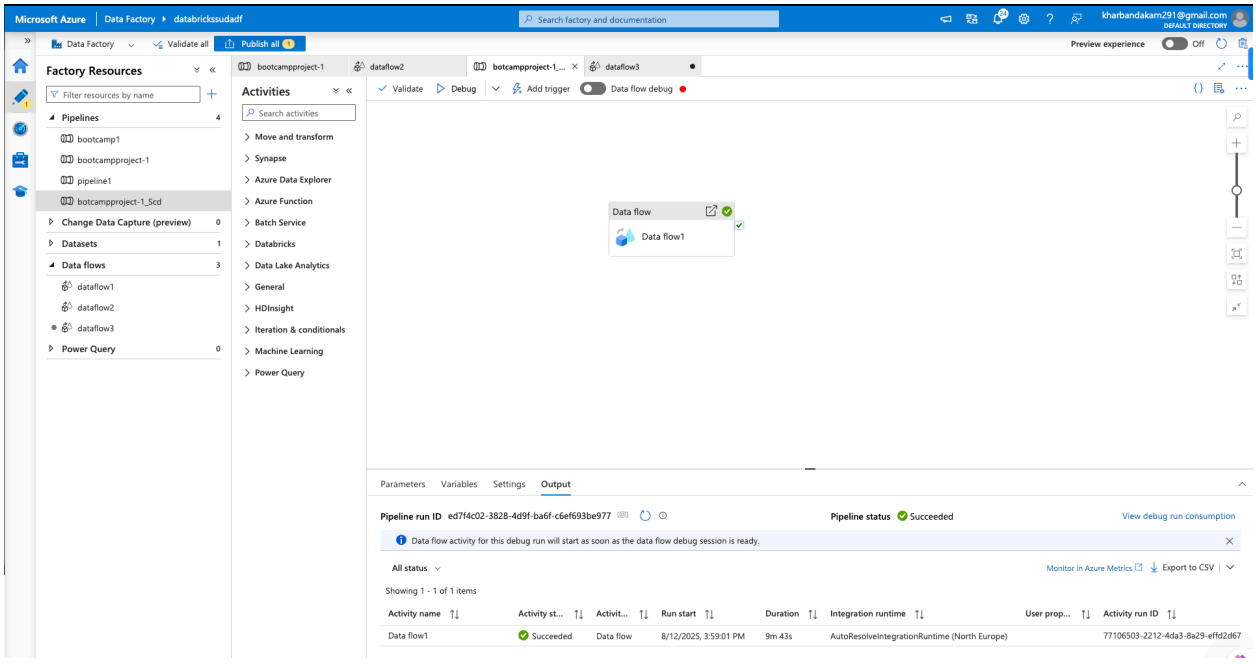
This source activity is used for loading the 5th csv file.



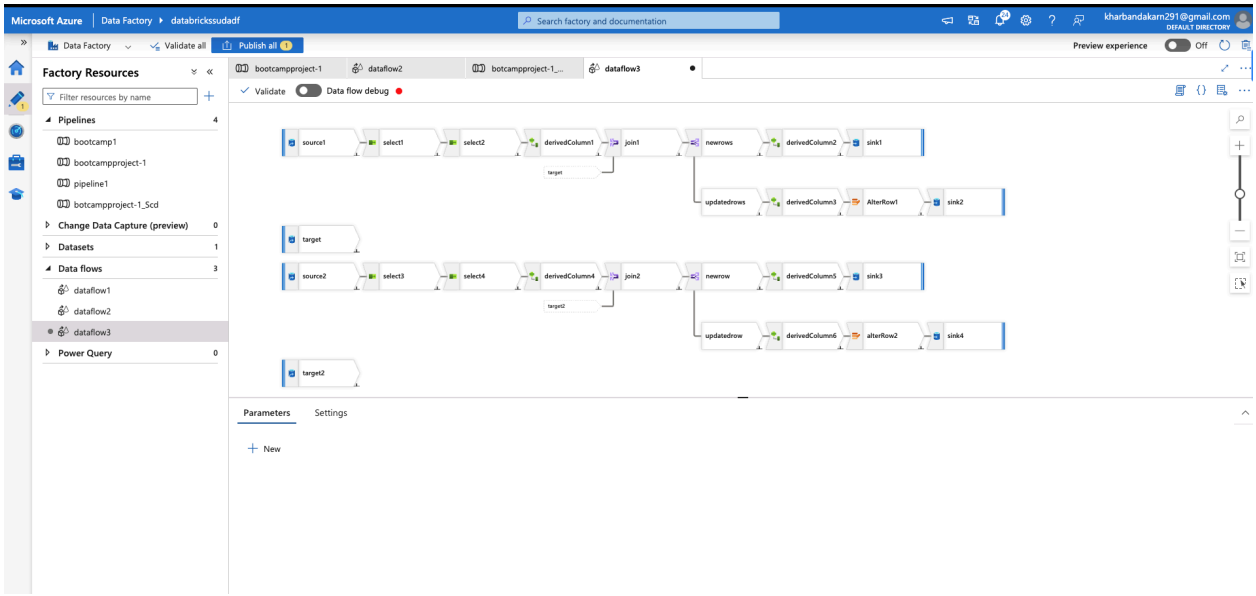
Now we are providing the path for the loans.csv file from where it has to loaded.



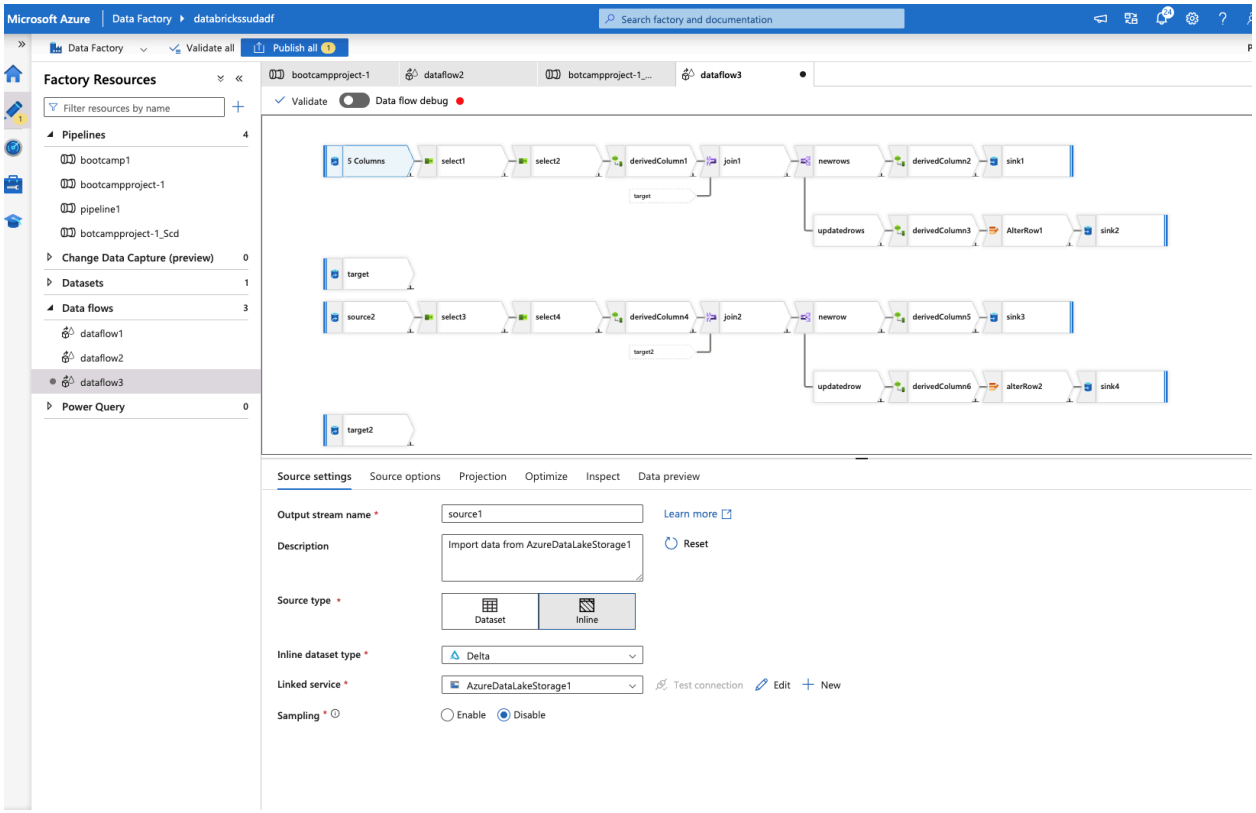
We are creating a new data flow in the new pipeline form where SCD type-1 will be implemented.



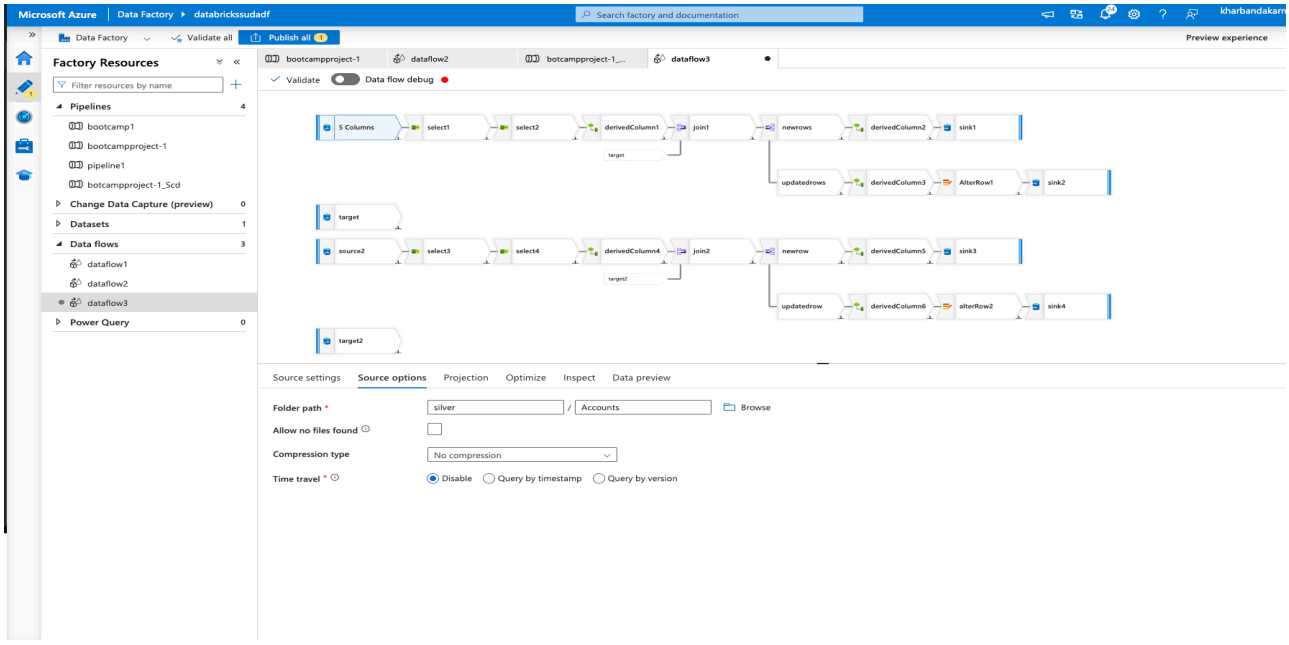
We are going to implement SCD type -1 in 2 of the 5 csv files so here is how the pipeline will look like inside the dataflow.



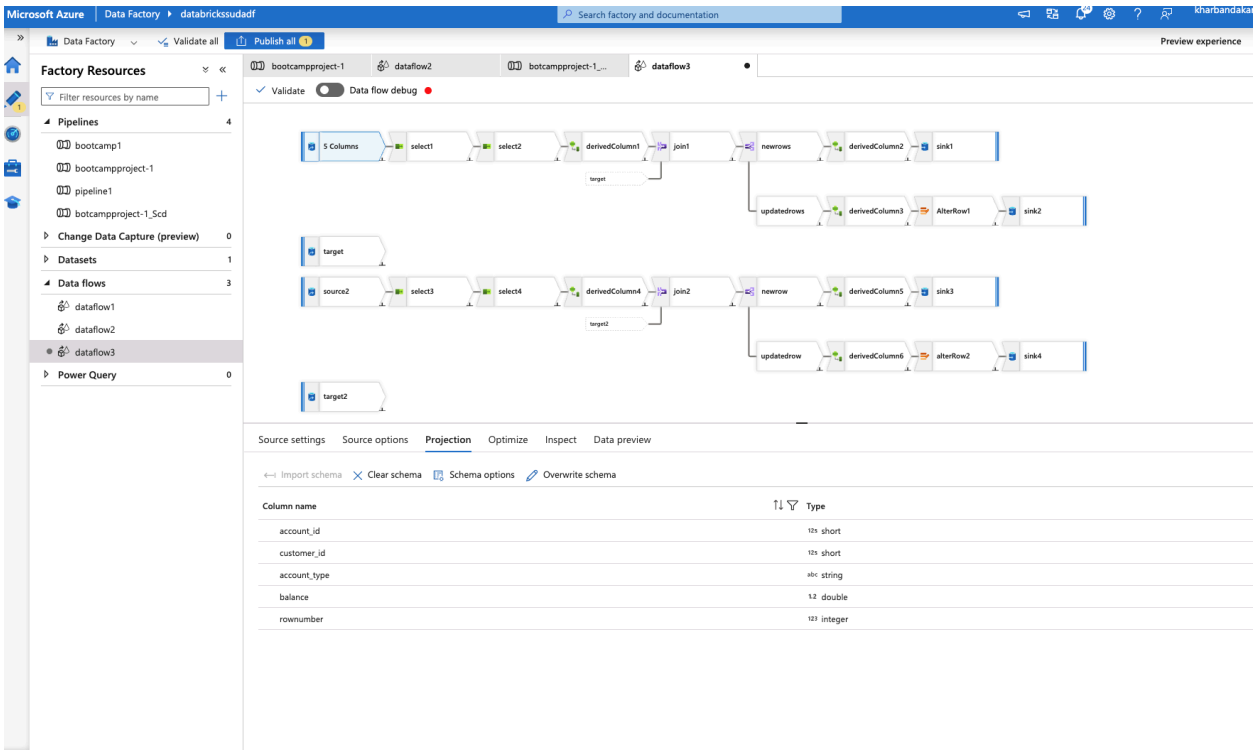
Source activity has been added that will loading data from delta format present in the silver container.



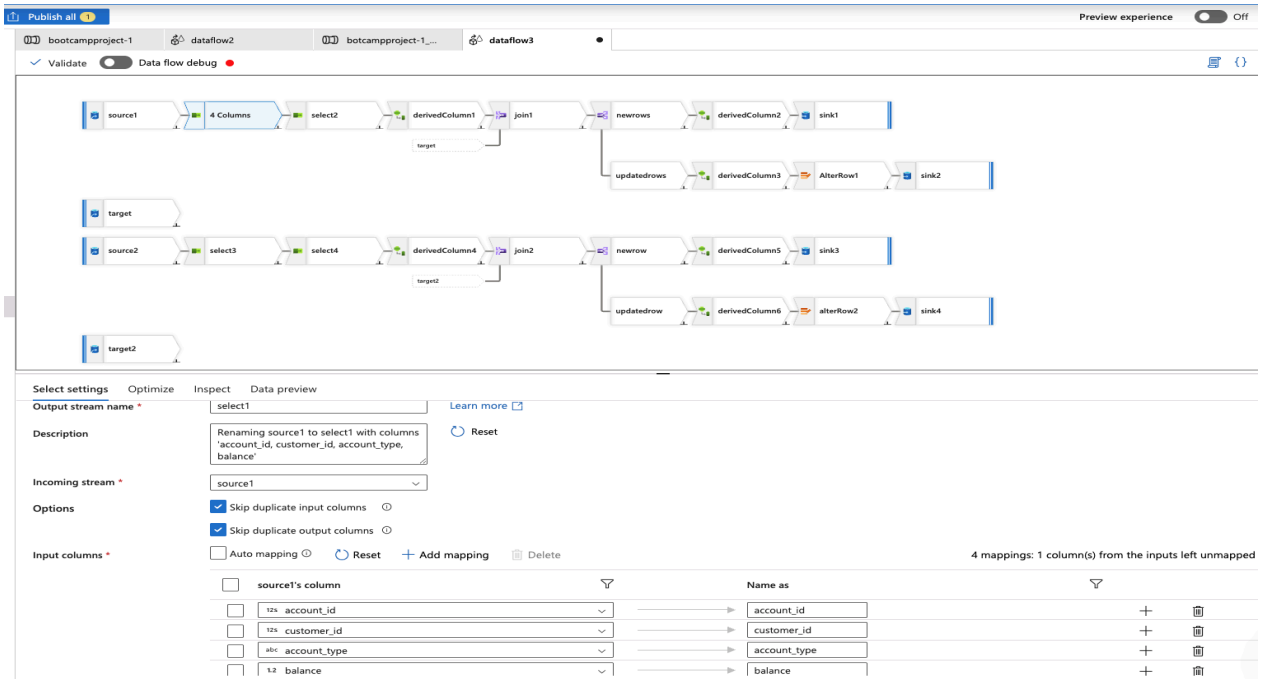
Here we are providing the path of file where it is stored.



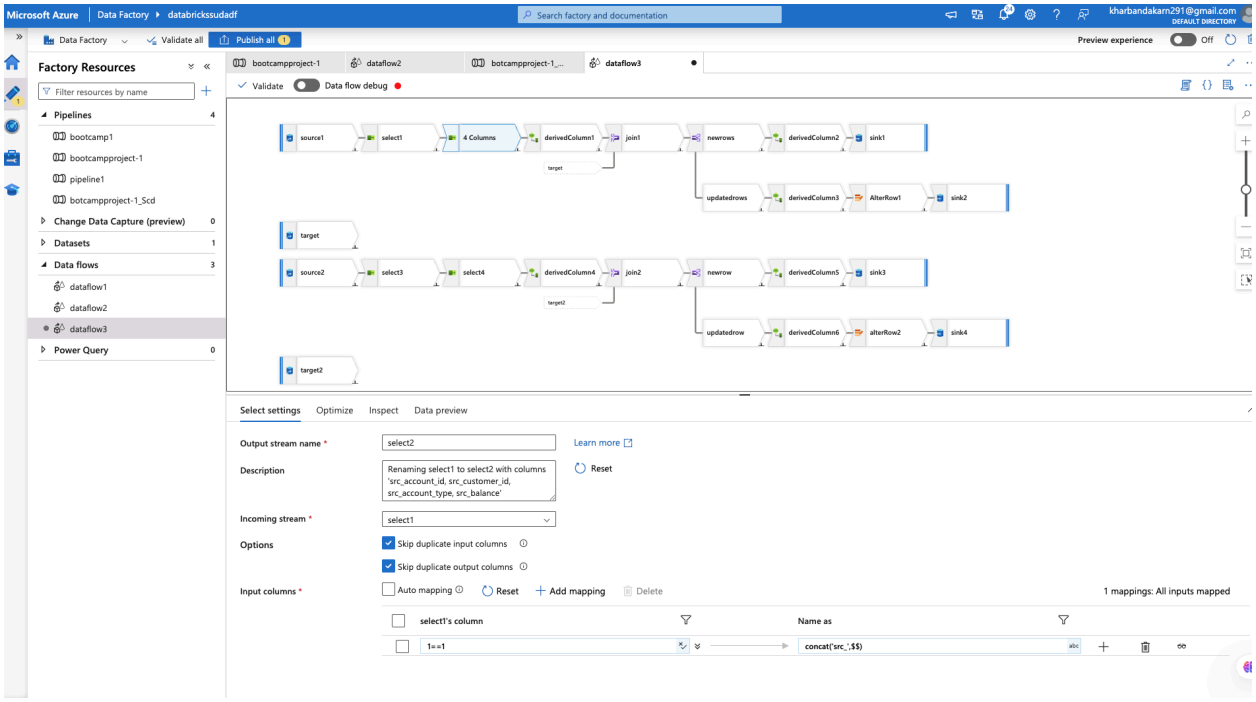
Here is how data will be coming in to SCD type dataflow from the silver container.



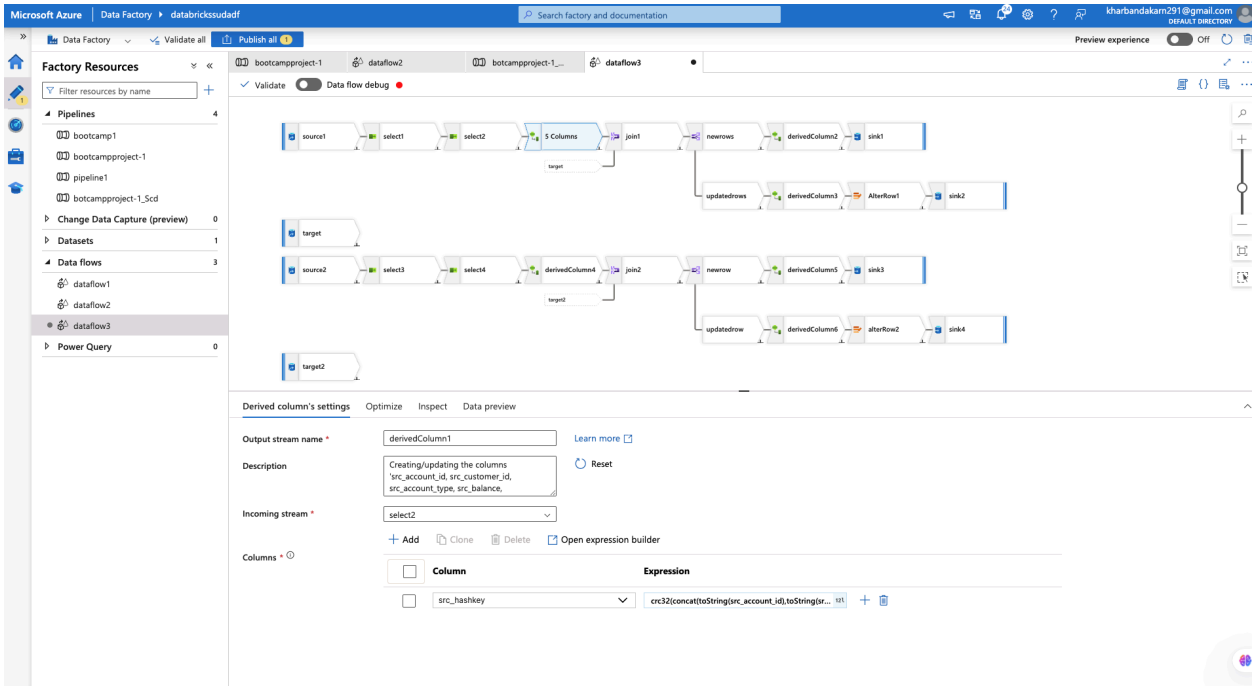
Now the select activity has been used for removing the unwanted column rownumber from the file.



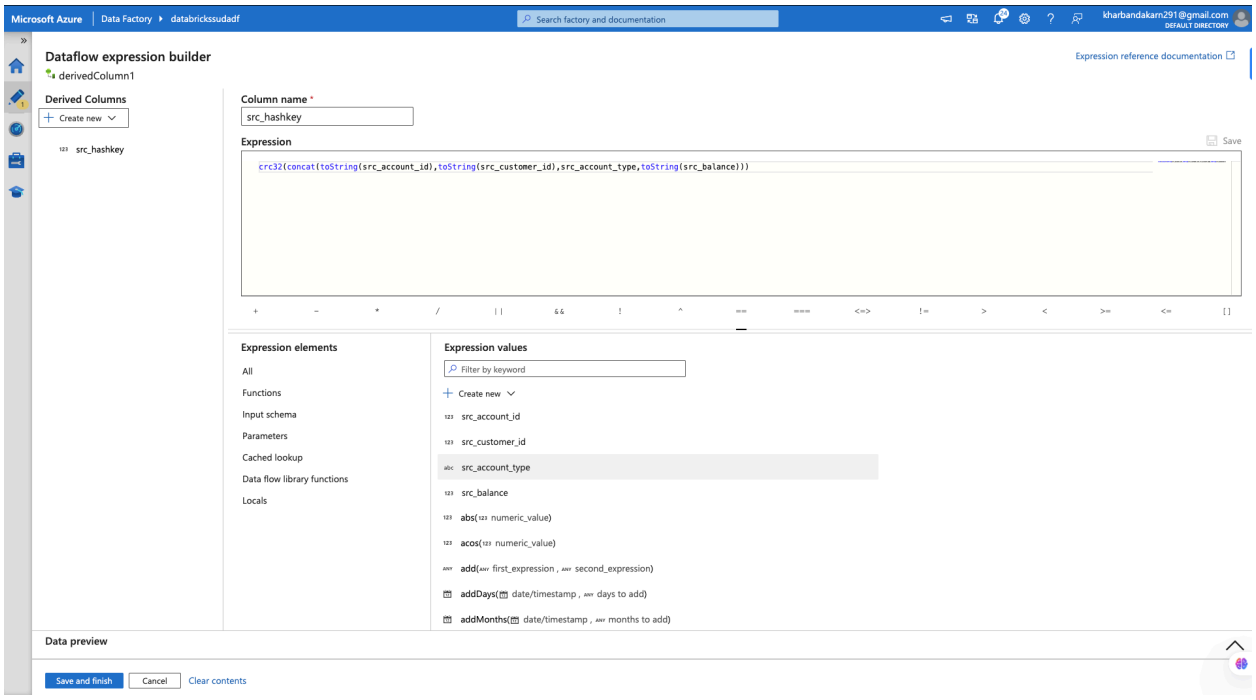
Now we will add “src_” in all the columns of the table.



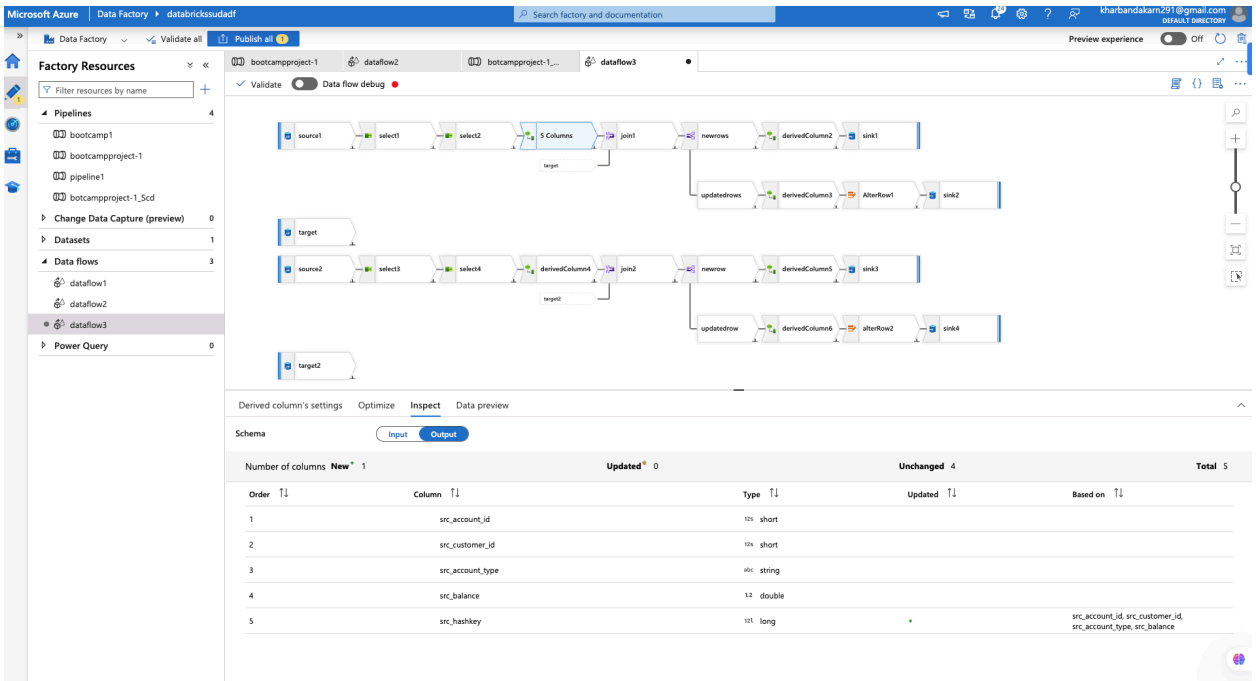
Here, using derived column activity, we are making a new column which will contain haskey. The haskey will made using all the column using crc32.



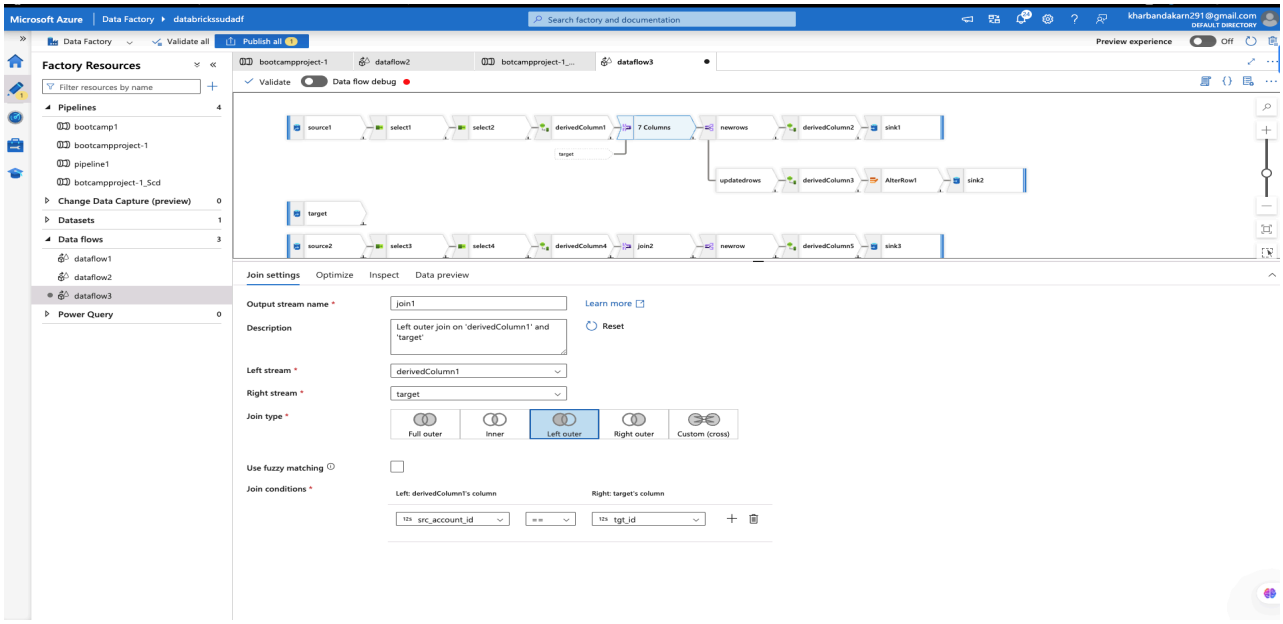
Here is the expression that we are using for making hash key.



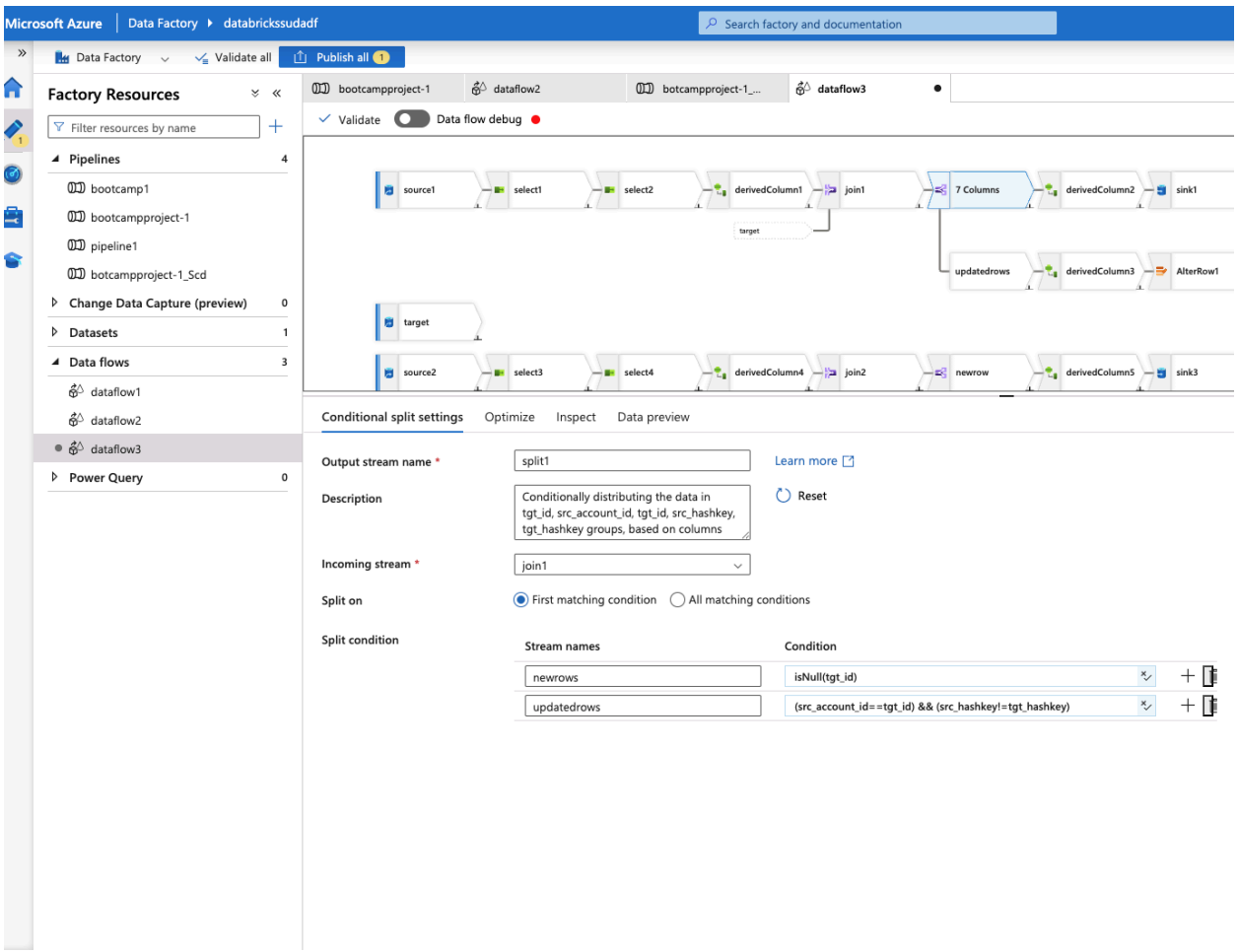
Here we are inspecting how the data will look like.



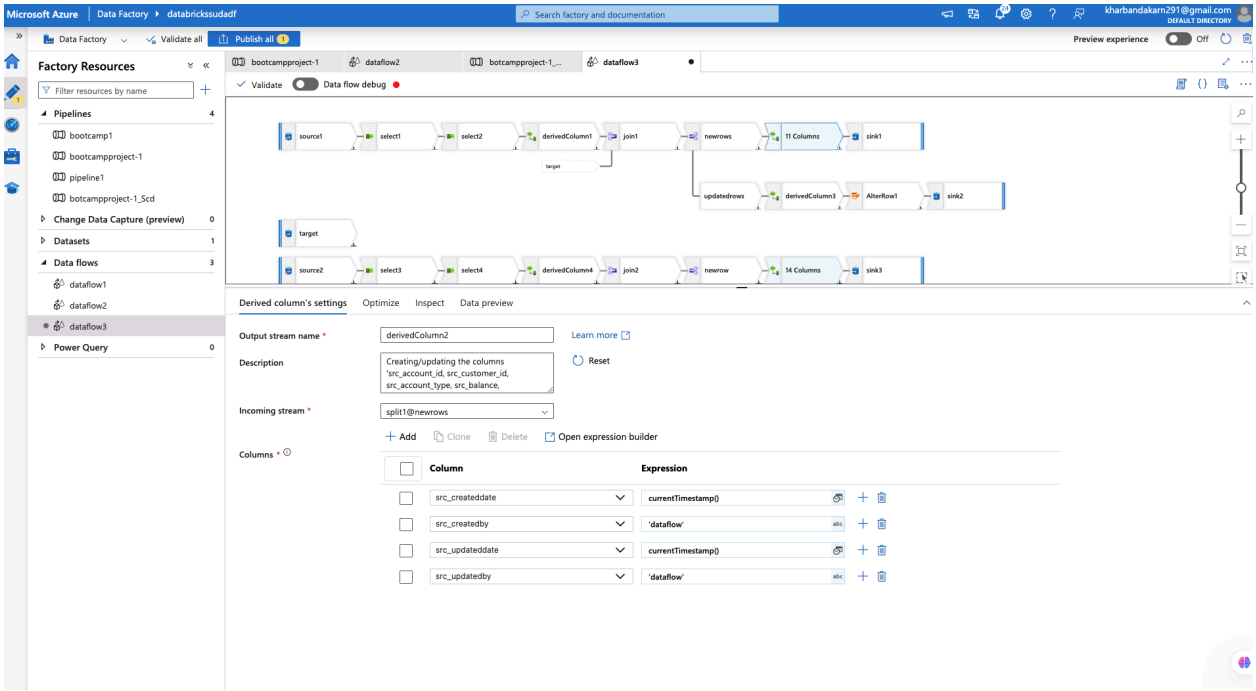
Now we are adding join activity where we will be joining two different tables. I.e the source and target(scd type-1) table using left outer join.



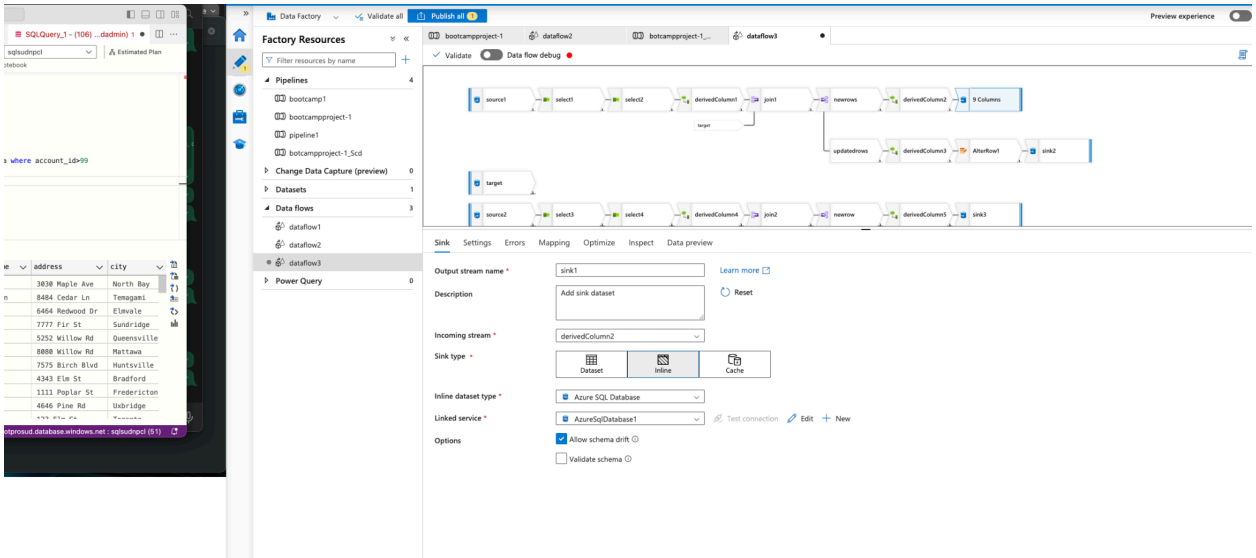
Here by using a conditional split we are splitting the data into 2 streams. One stream will contain the rows that are newly inserted, while the other stream will contain the rows that are updated.



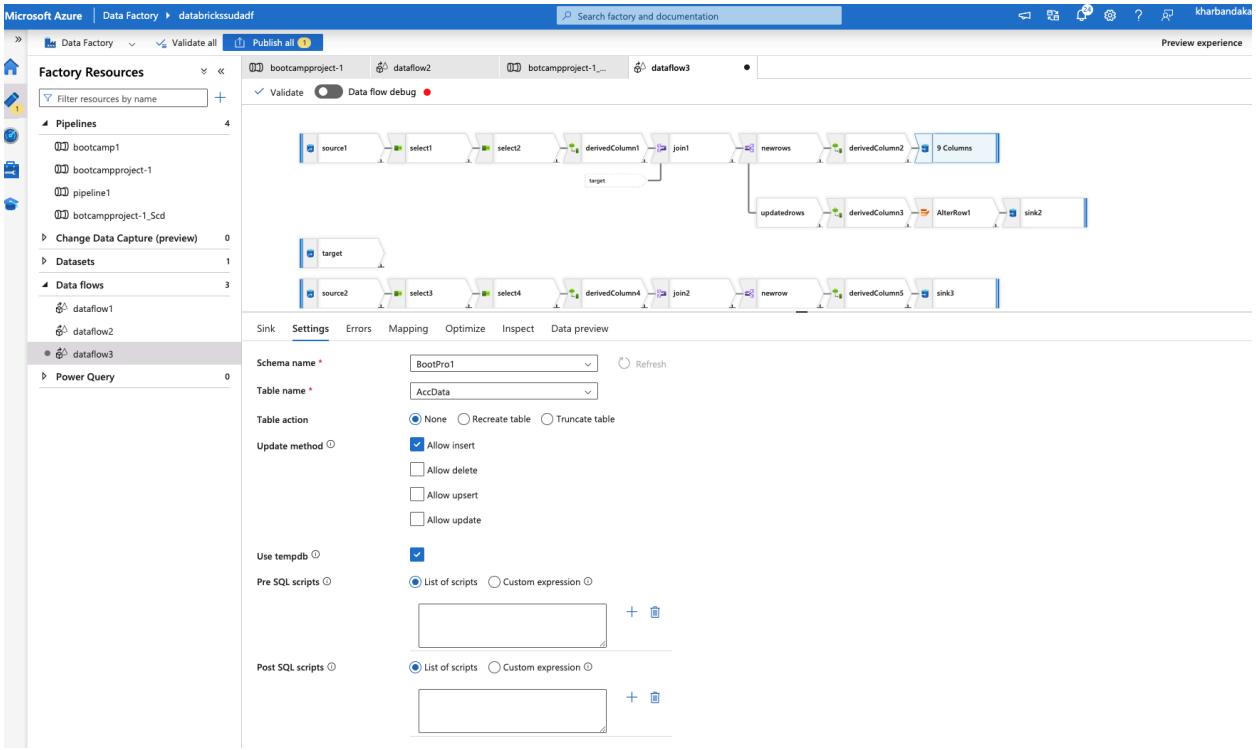
Now using the derived column activity we are adding new column so we can map the data into the SCD type table.



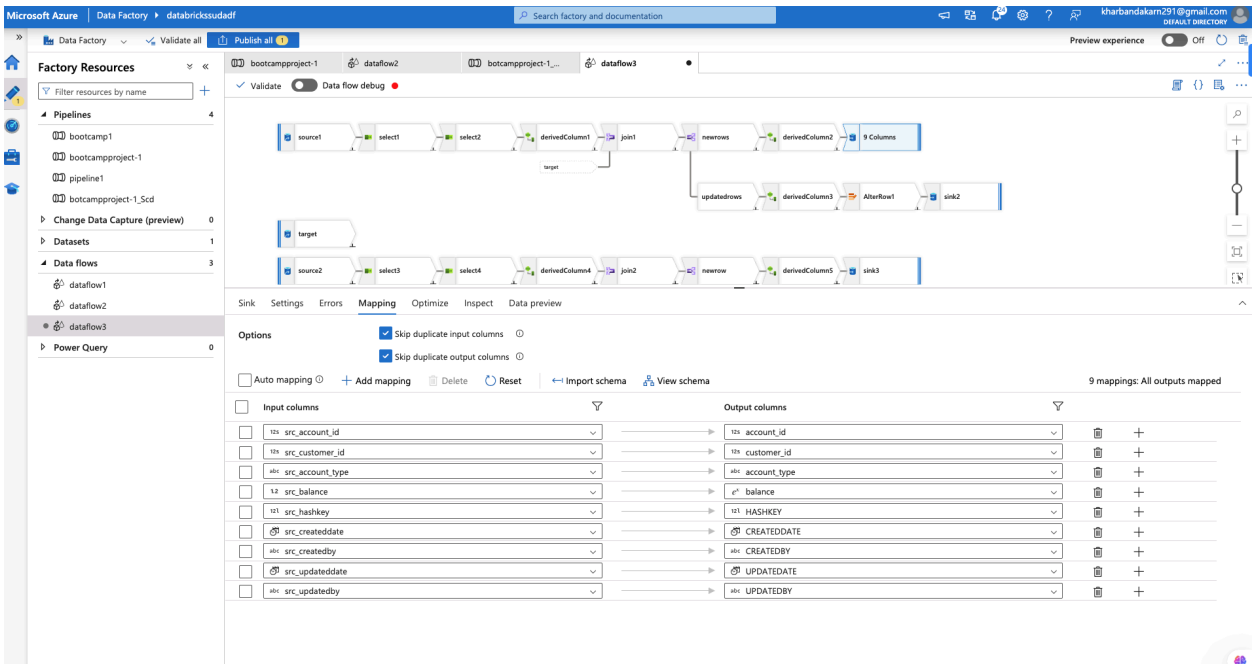
Now we will be using a sink activity for saving the data in SQL database. Here we will do the following configuration for implementing sink.



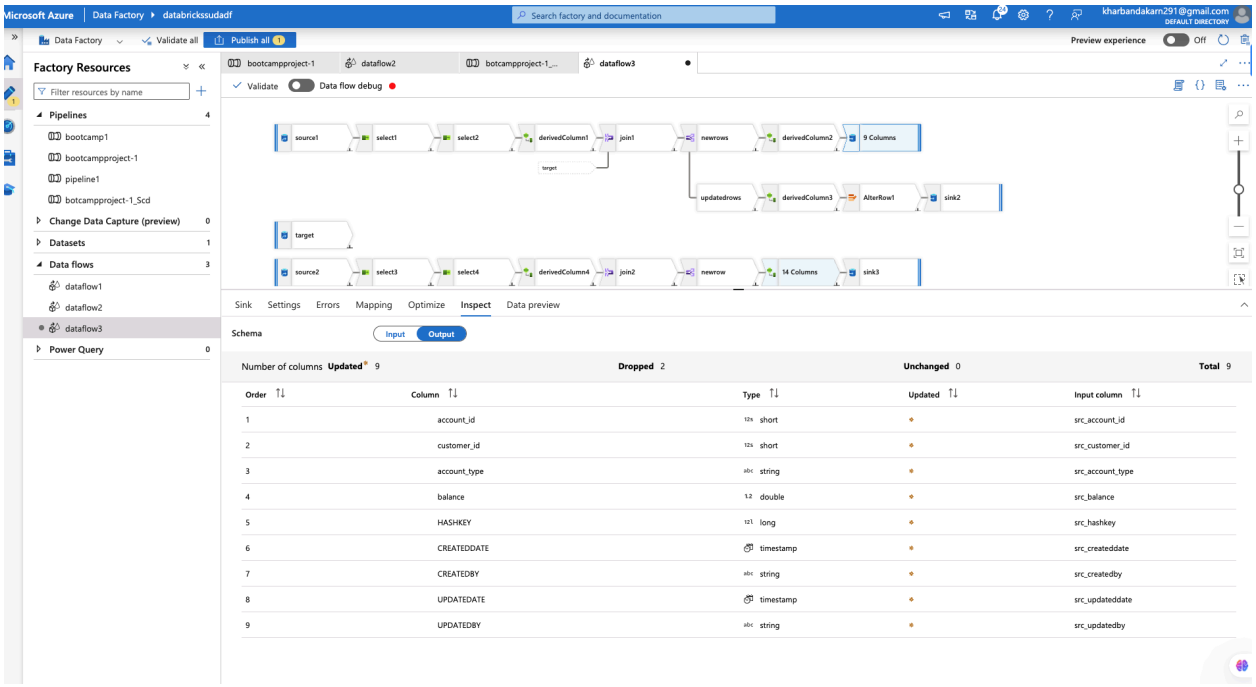
In the setting tab select insert because here we will be inserting new data only, and select the schema name and table name.



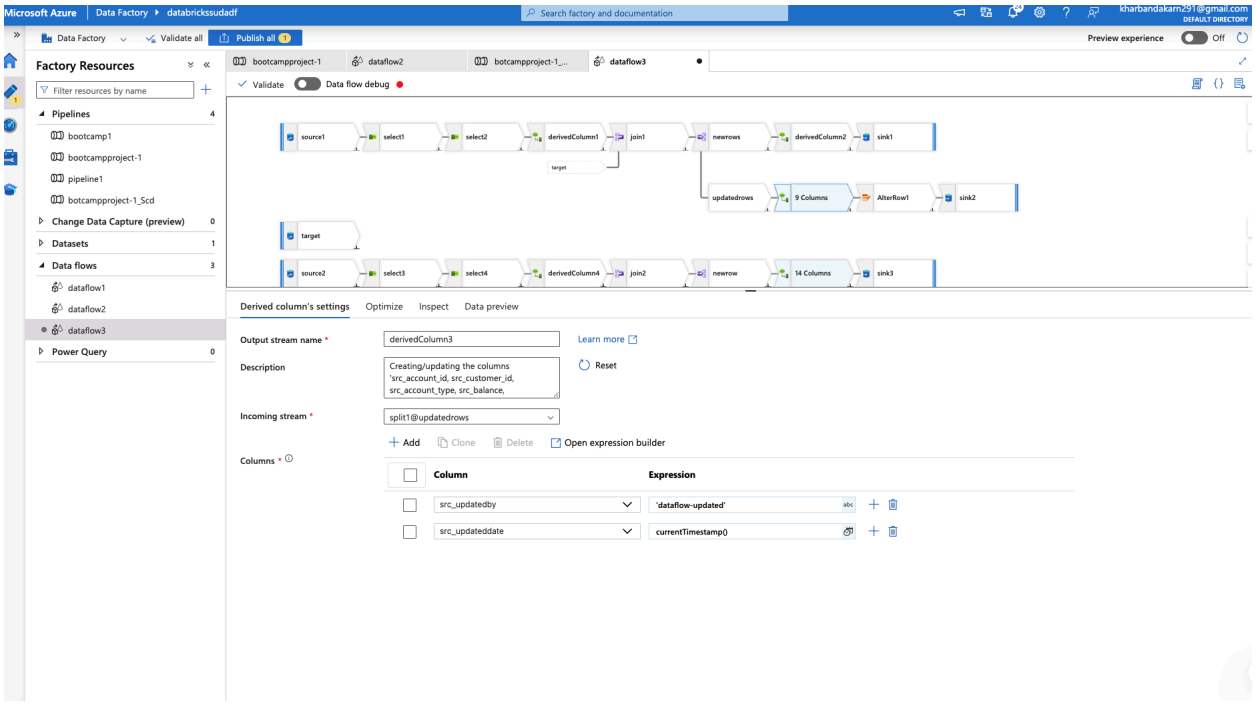
Do the following mapping by aligning the source columns with the target scd type columns.



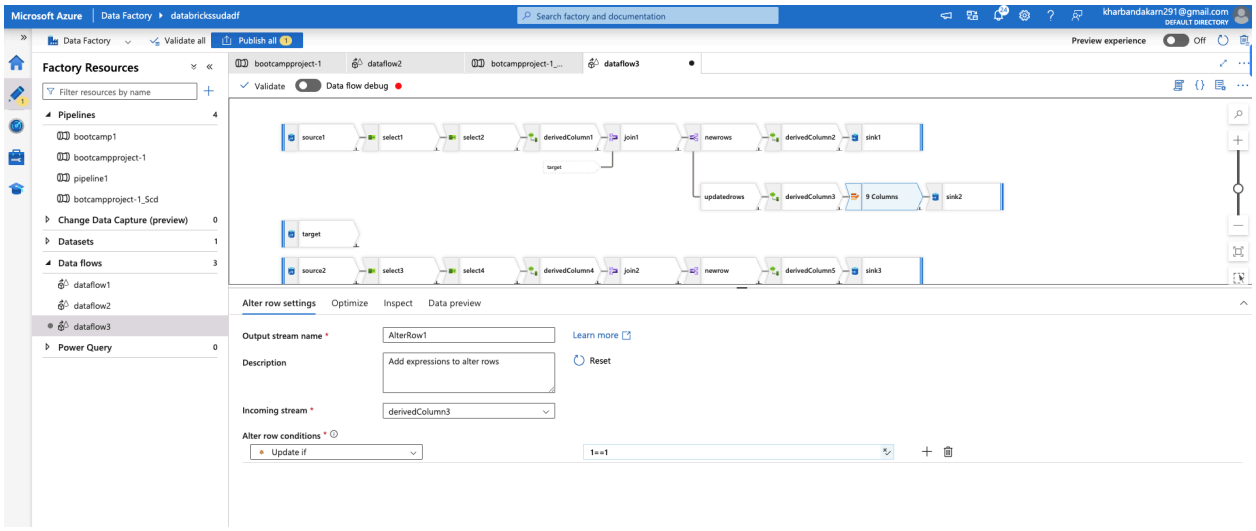
Here we are inspecting the data before it is saved into the SQL database(Gold Layer).



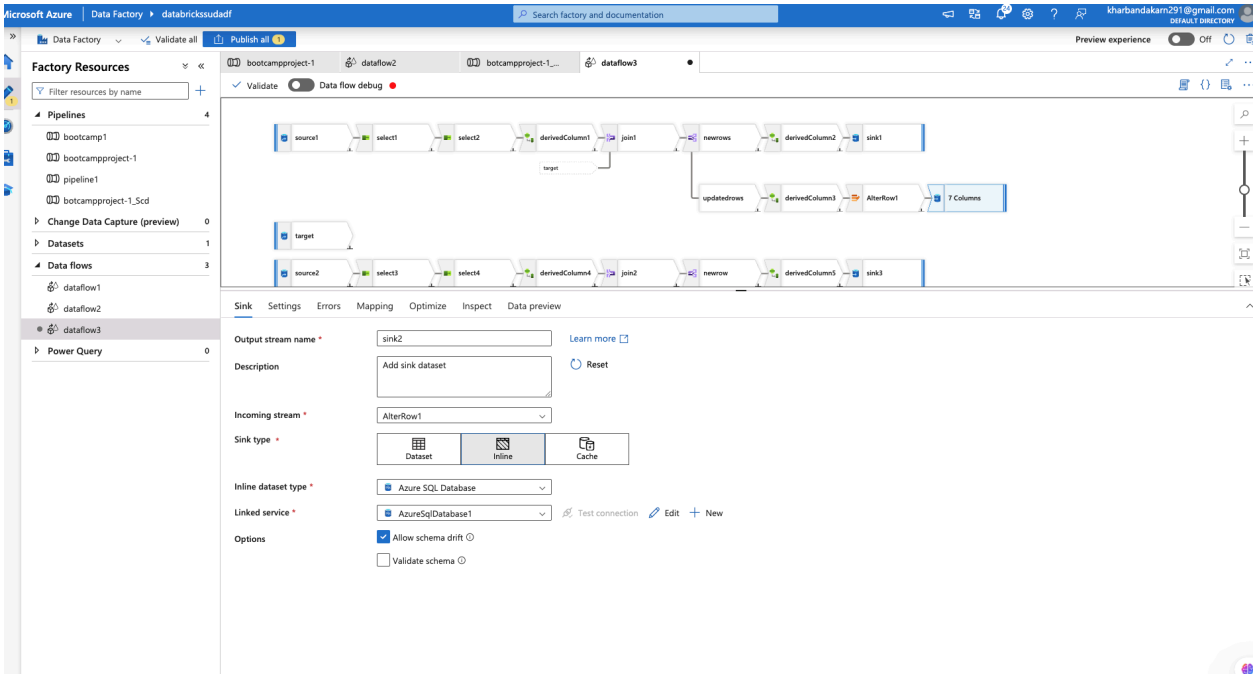
Now we add derived column activity after the updaterows stream. Here we will adding only updated by and updateddate column since we want add the recent updated time and person.



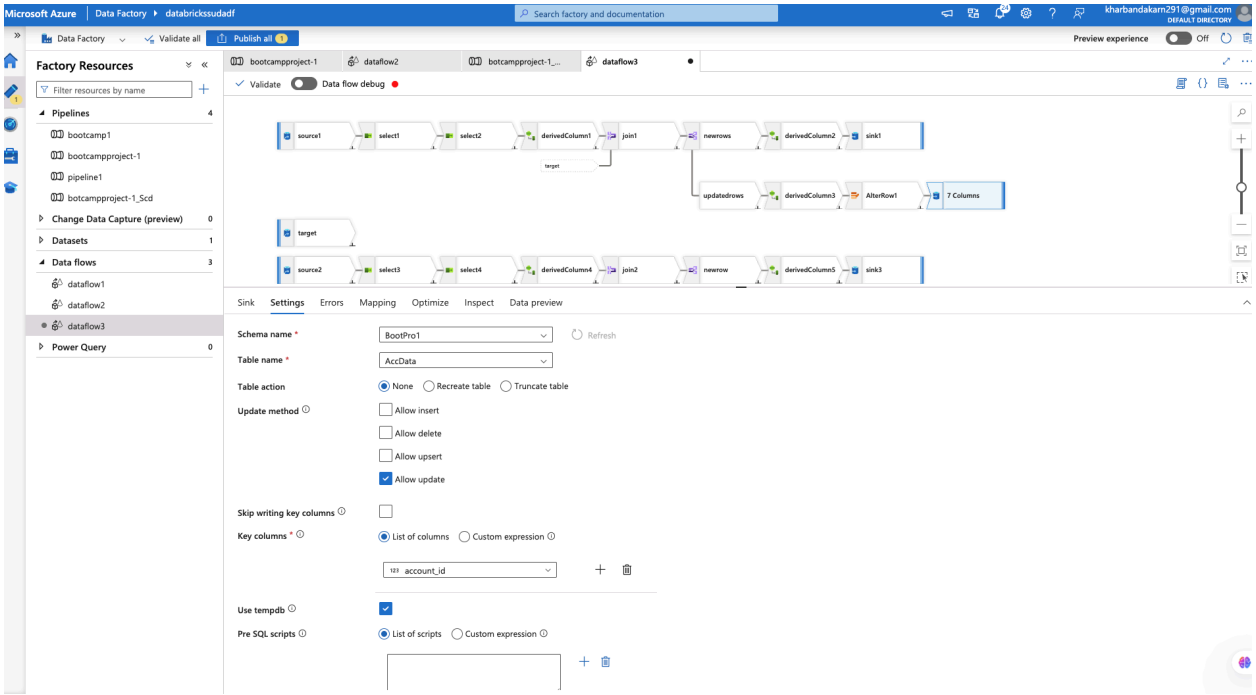
Here we will be alter row activity since update will be used in the sink activity.



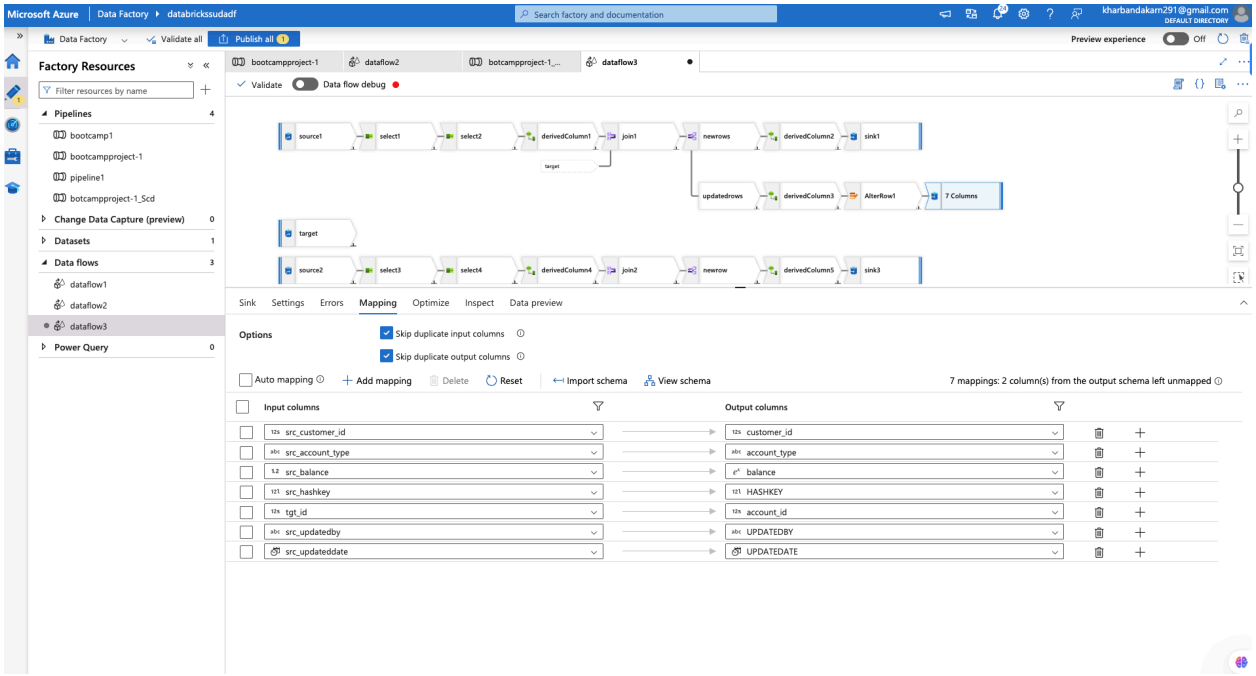
Here is the sink activity that will be sinking the updated row in the table.



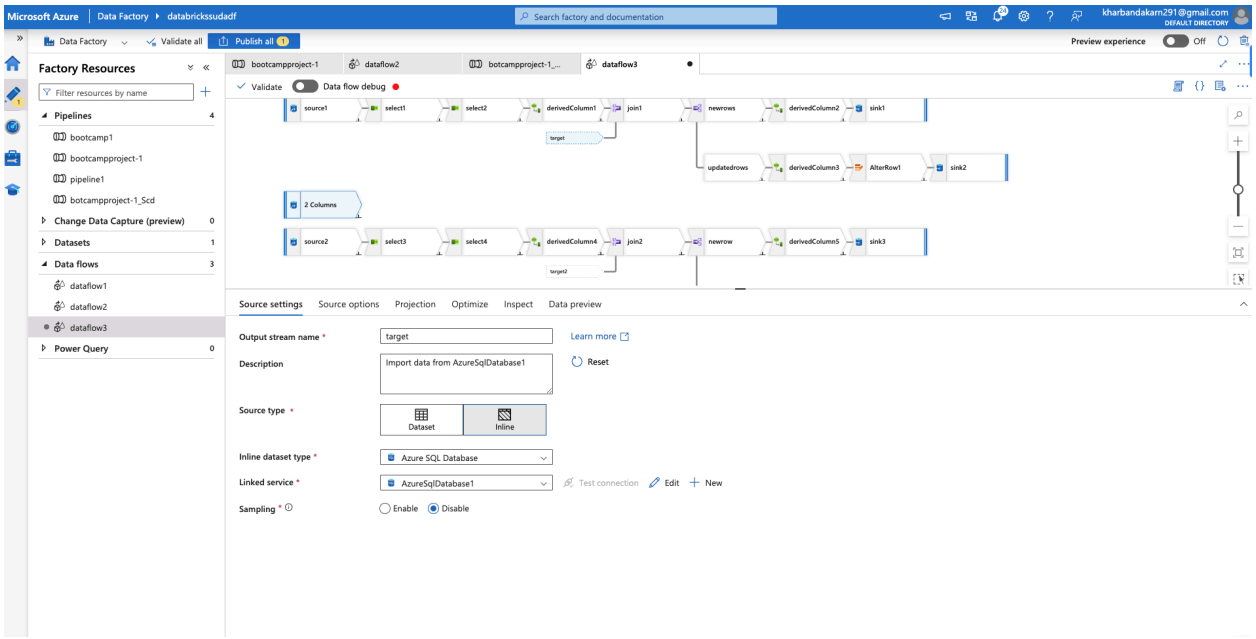
Add the schema name and the table and in the update method, select allow update.



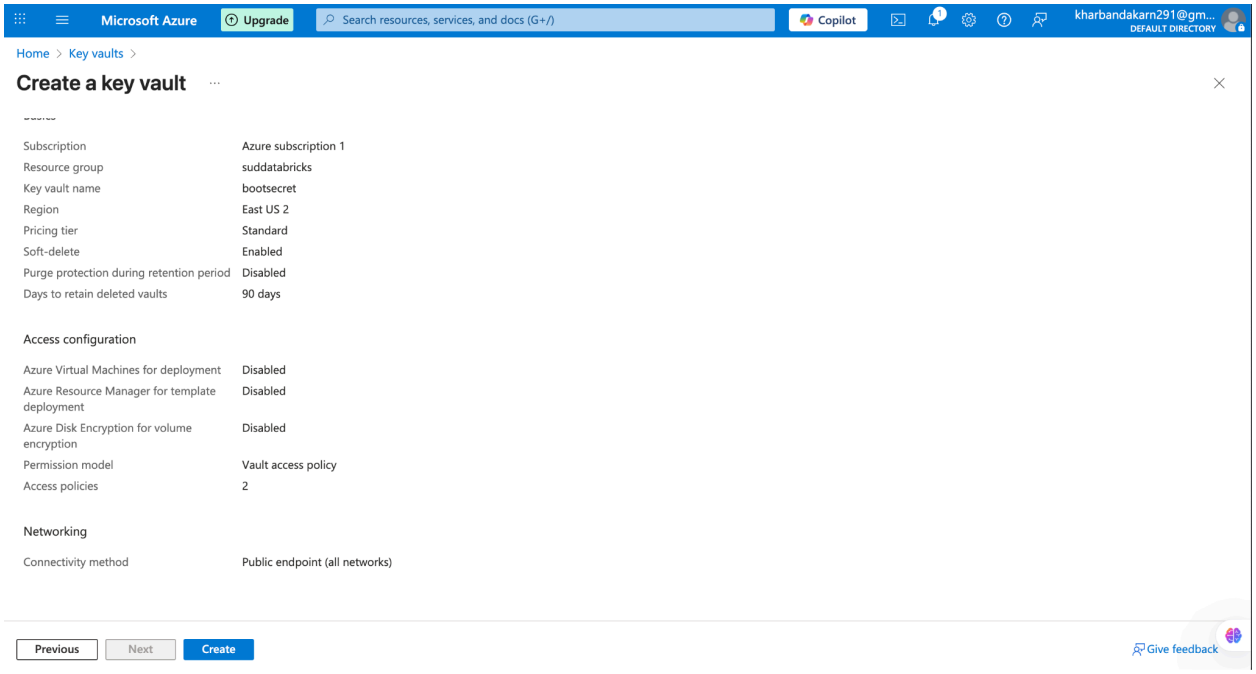
Do the following mapping for the updating the SCD type table. Note- Here we will remove createdby and createddate while mapping and update the old hashkey with the new hashkey.



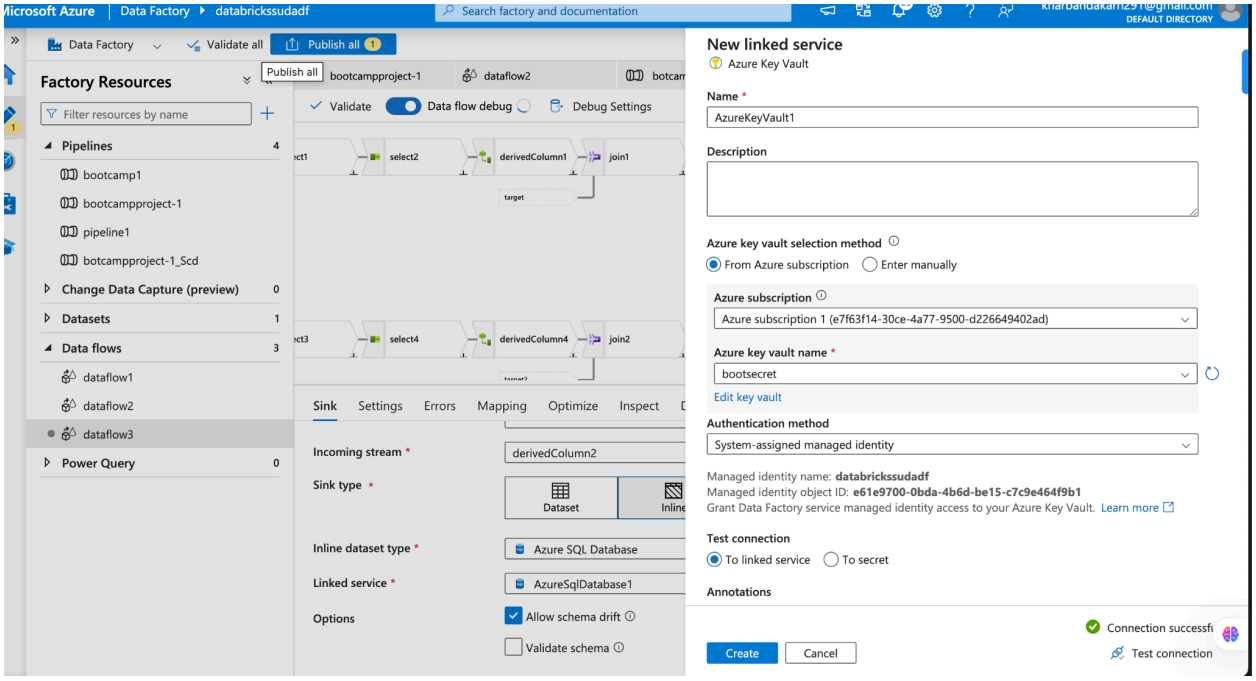
Here is the target activity for where the SCD type data will be loaded into the pipeline. Do the following configurations.



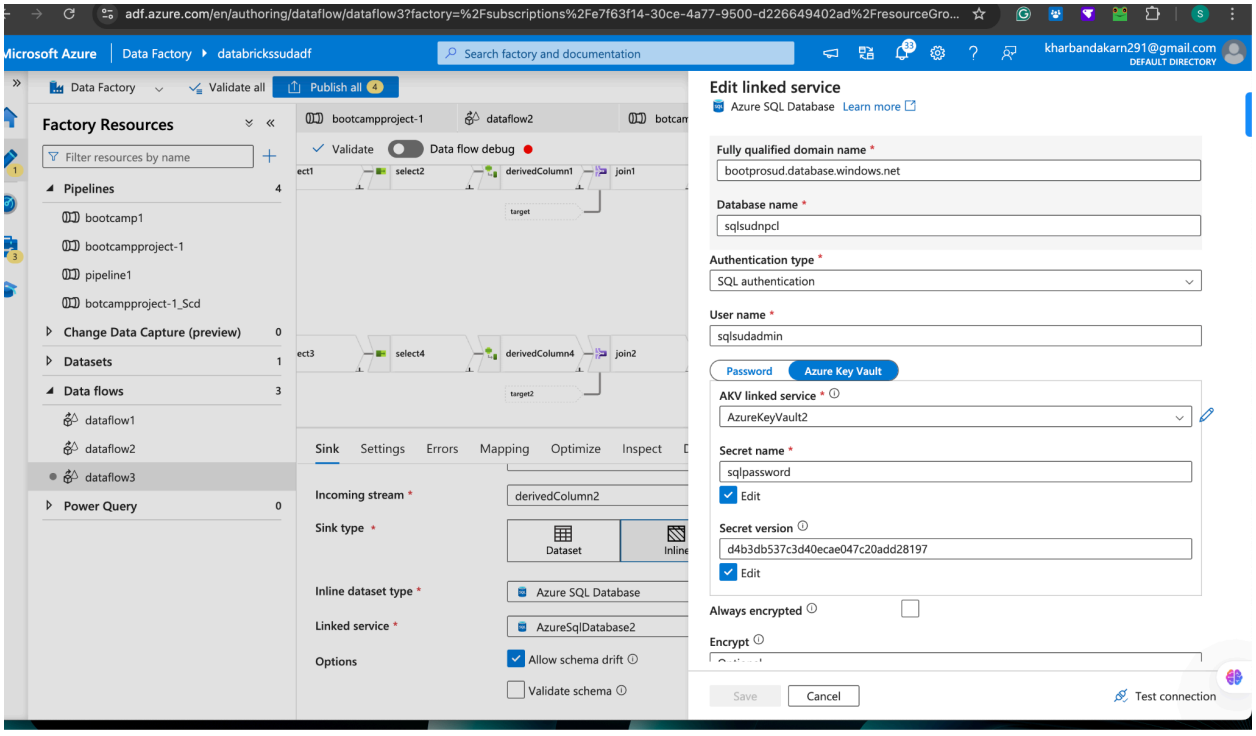
Creating a key vault for storing the password of the database.



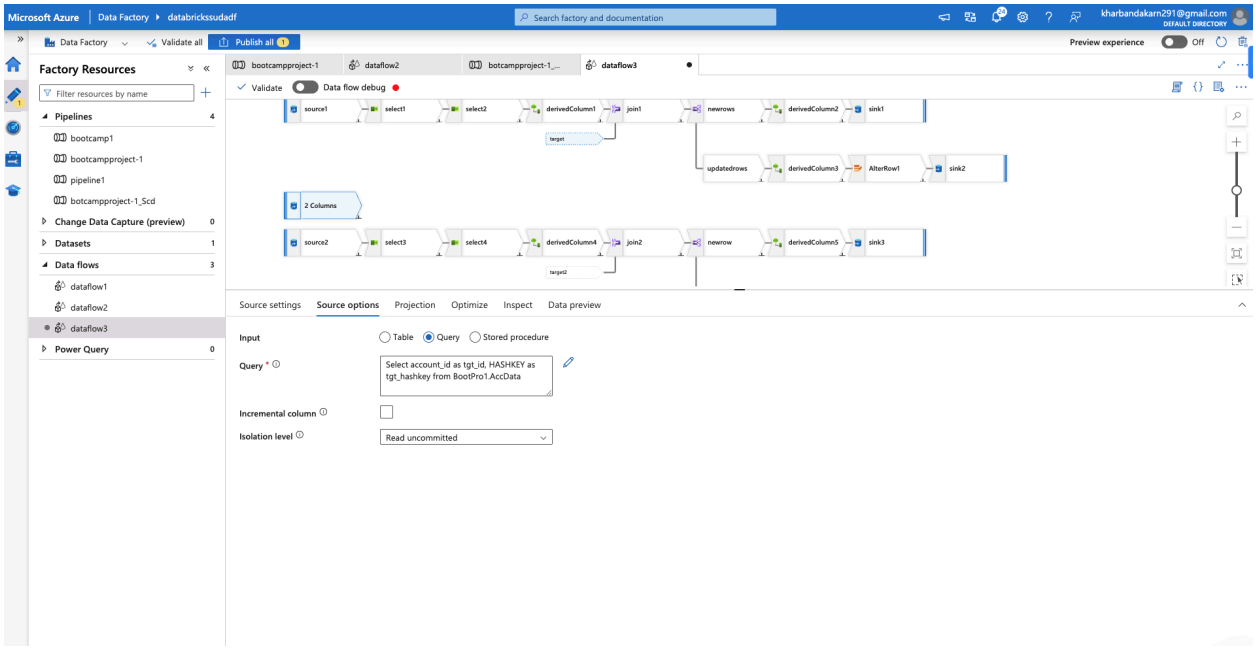
Creating the link service for connecting the key vault.

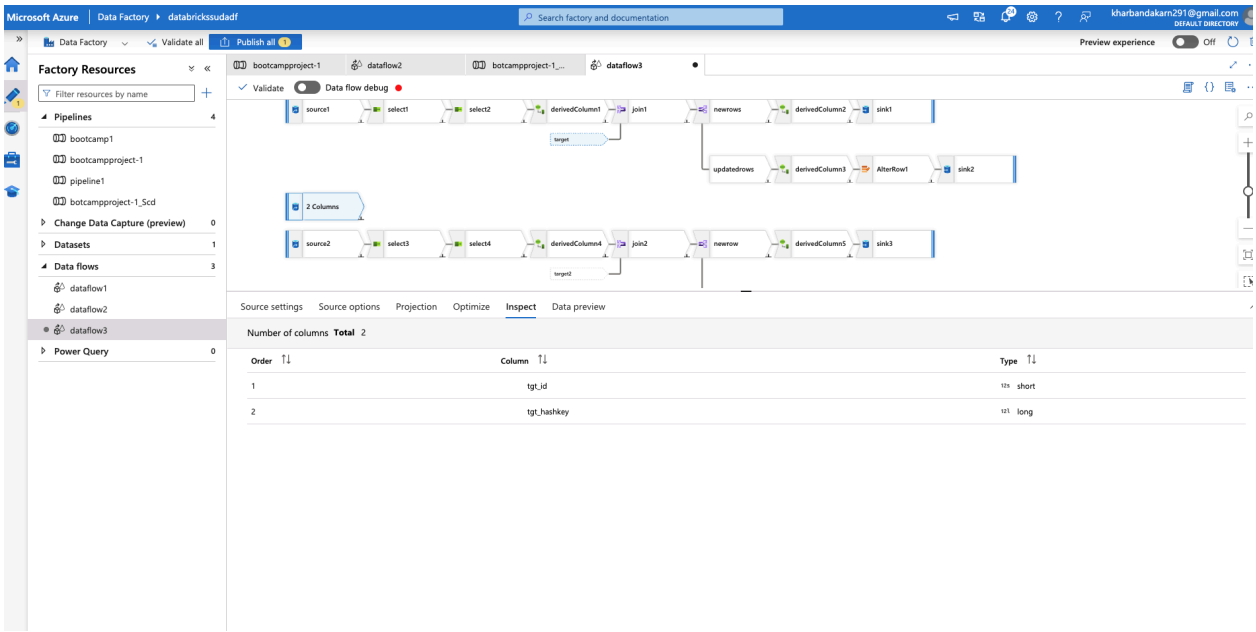


Now we have to select the linked service and secret name for the key vault.

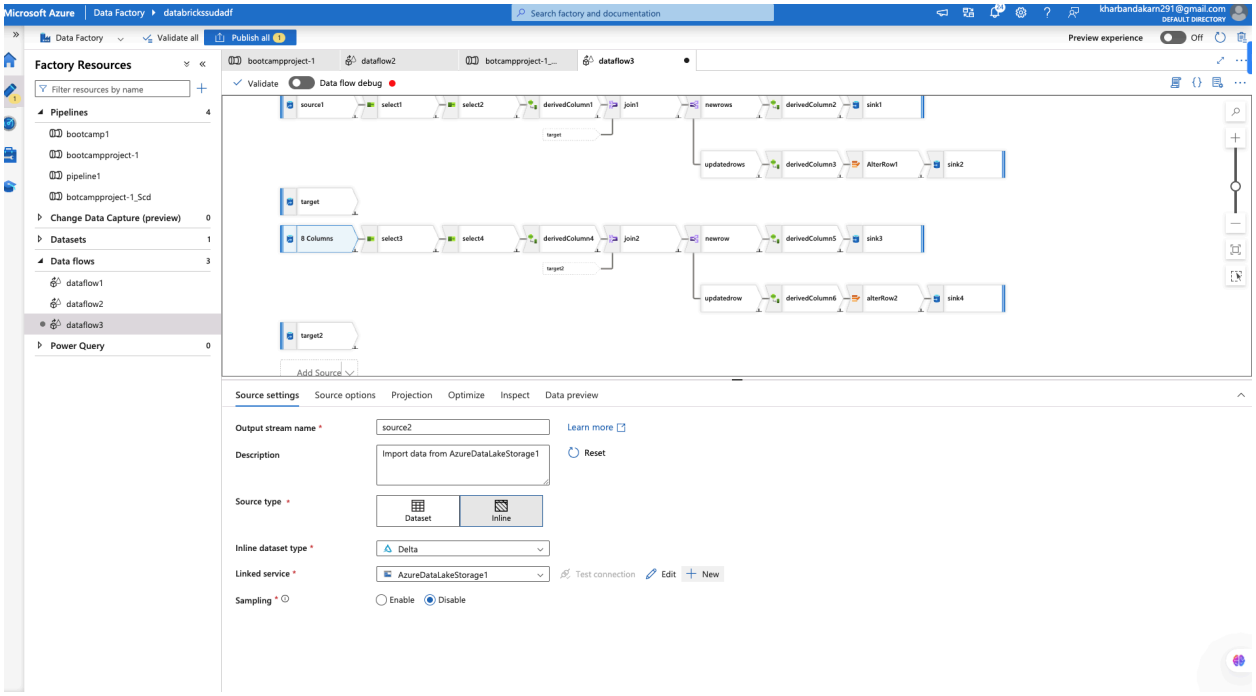


Use the following query to get the data into the table. Here, we will be fetching only the primary key and the hashkey column.

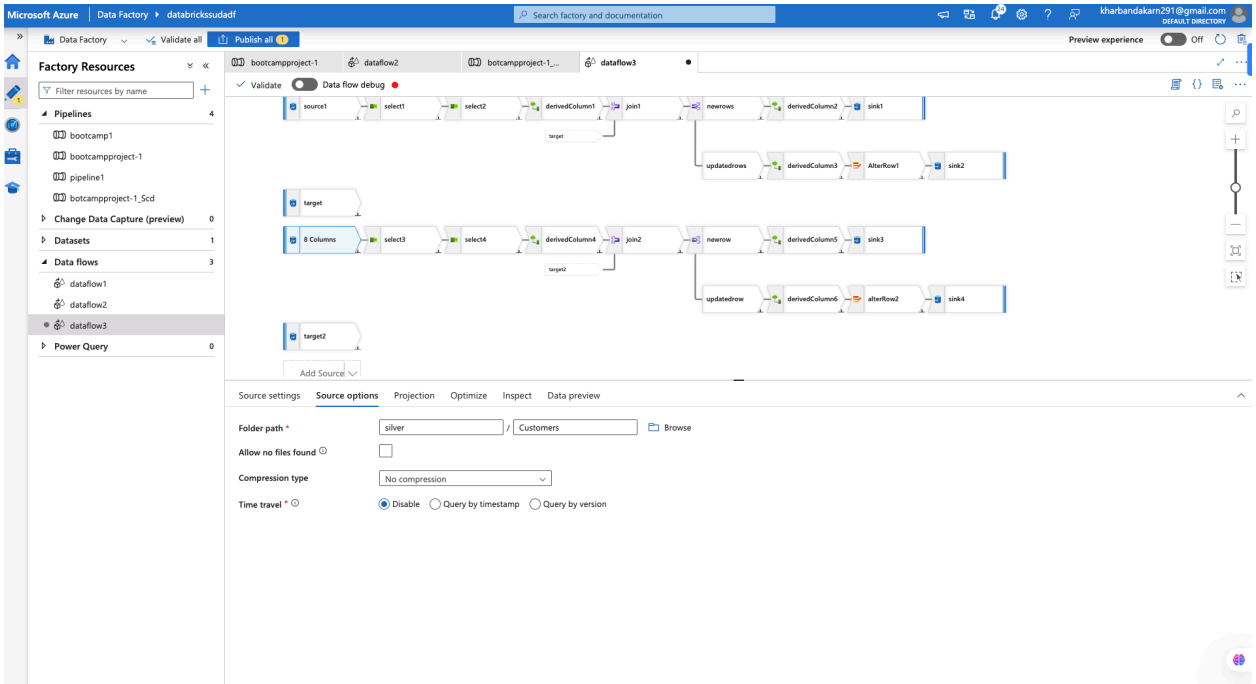




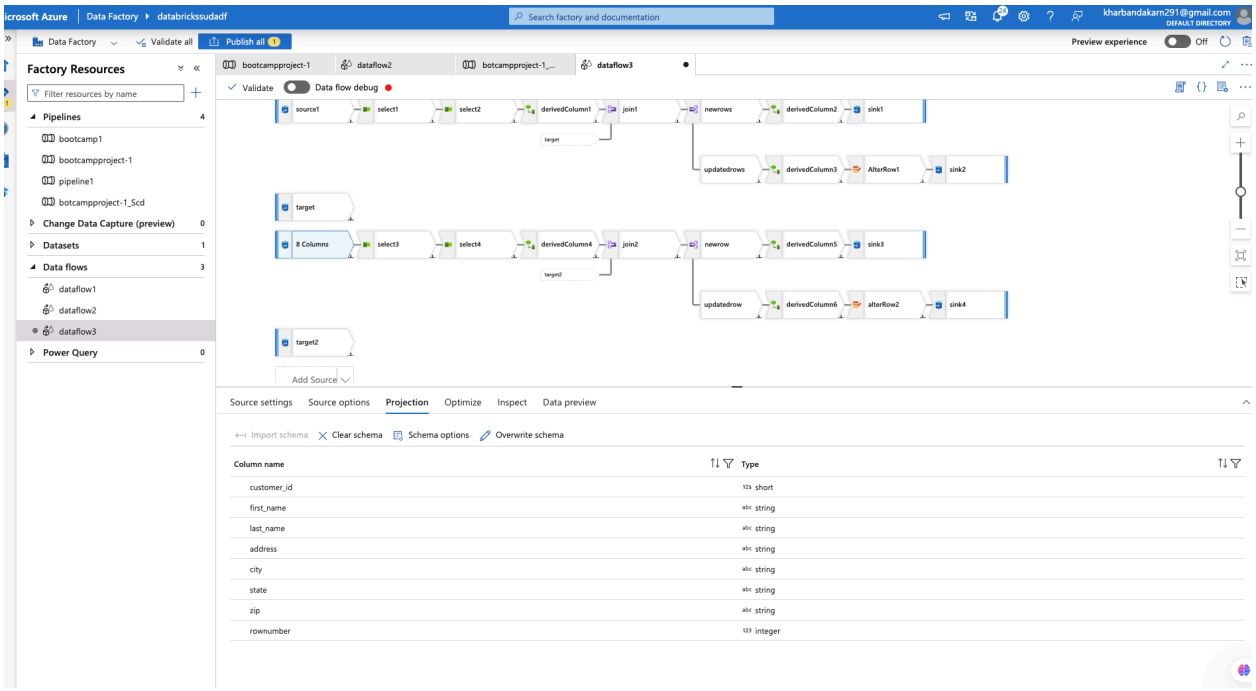
Now the same will be implementing scd type-1 for another csv file named customers. Do the following configuration for loading file into the pipeline from silver layer.



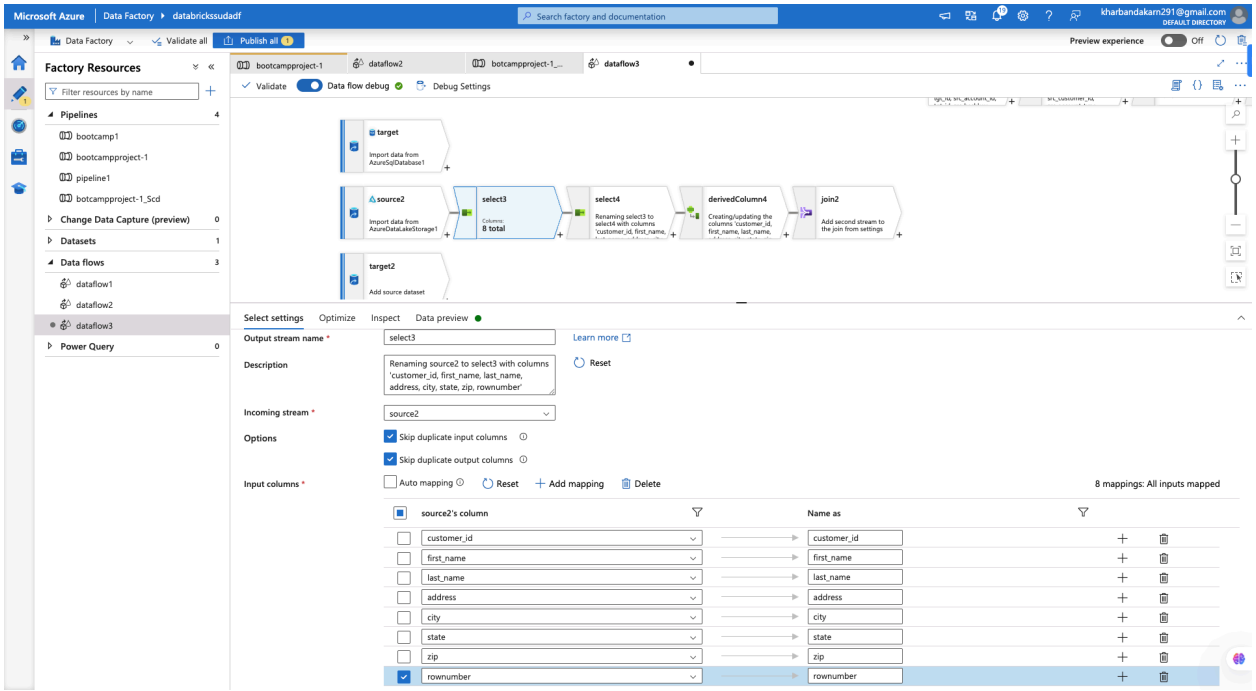
Here provide the path from where data has been loaded into the pipeline.



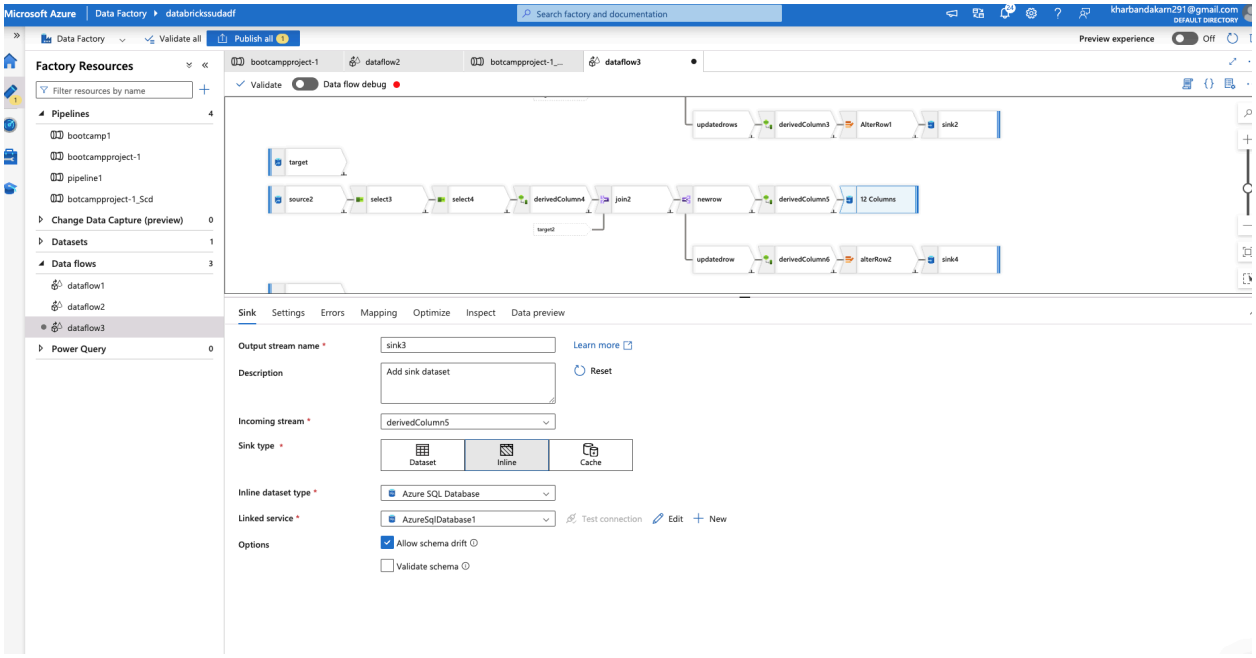
Now we will import the schema to check the columns.



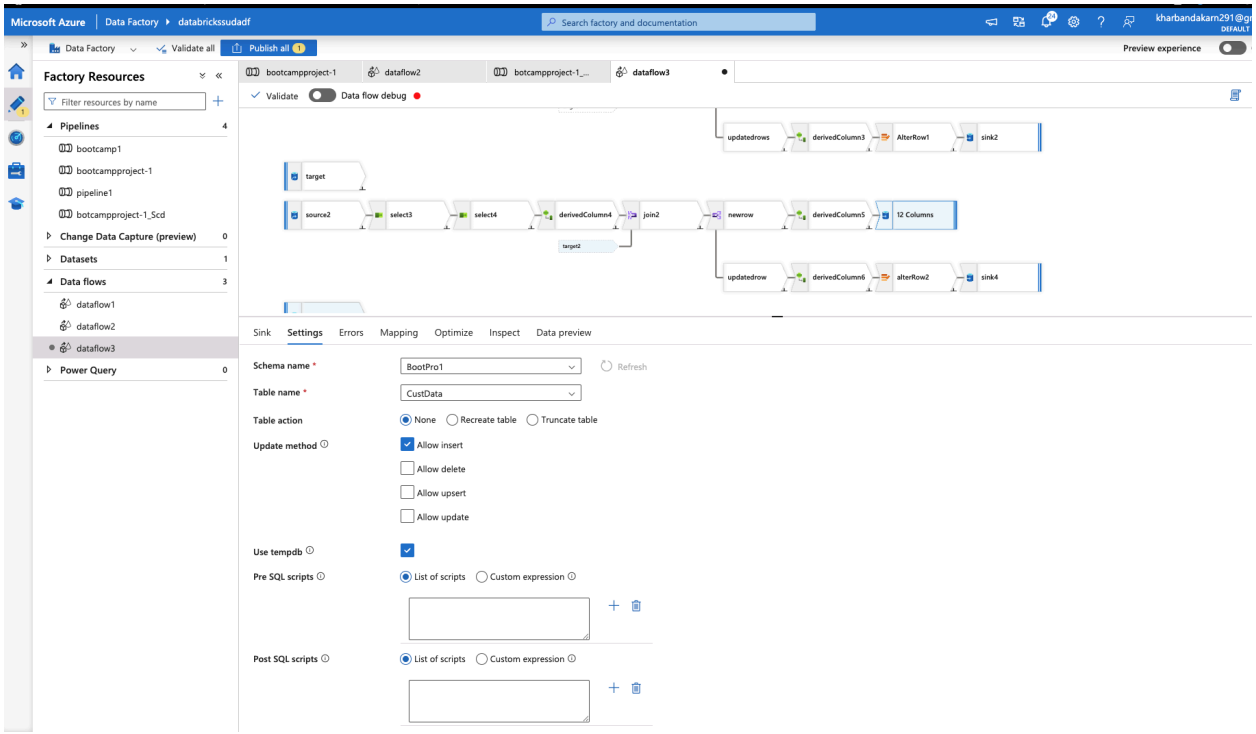
We are adding select activity to remove the unwanted column rownumber



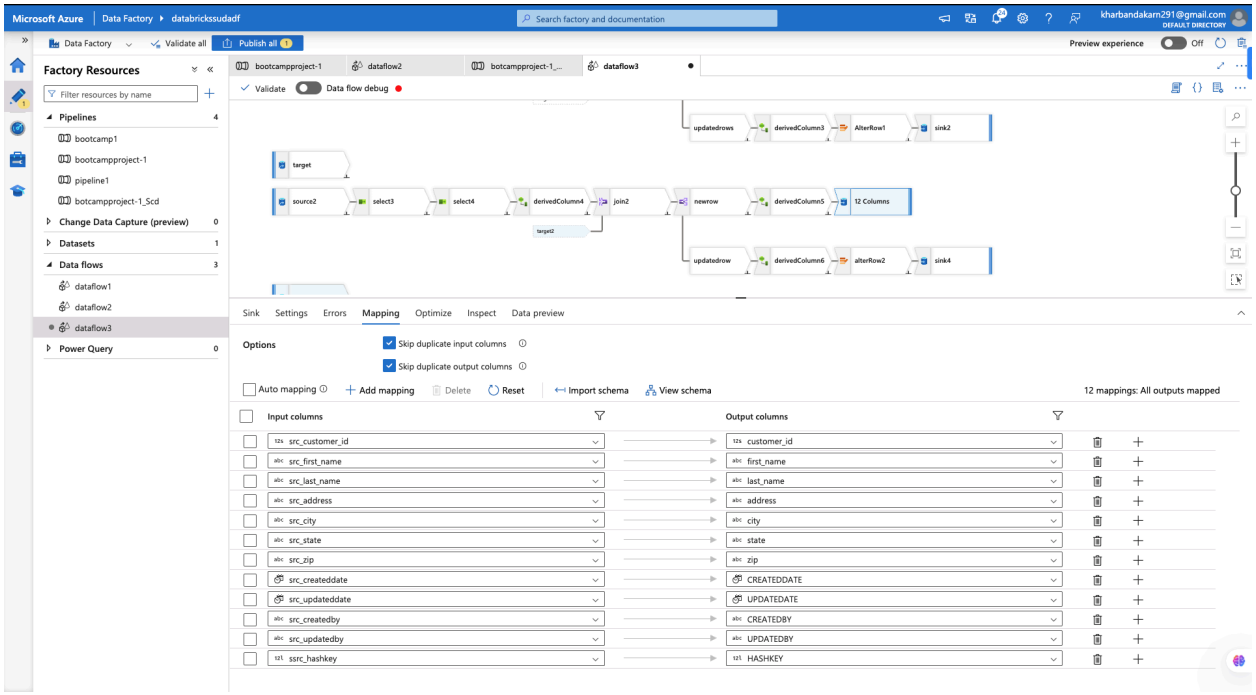
We will be implementing scd type-1 in the same way as we have implemented for the above CSV file there will be slight change in the mapping part of the sink.



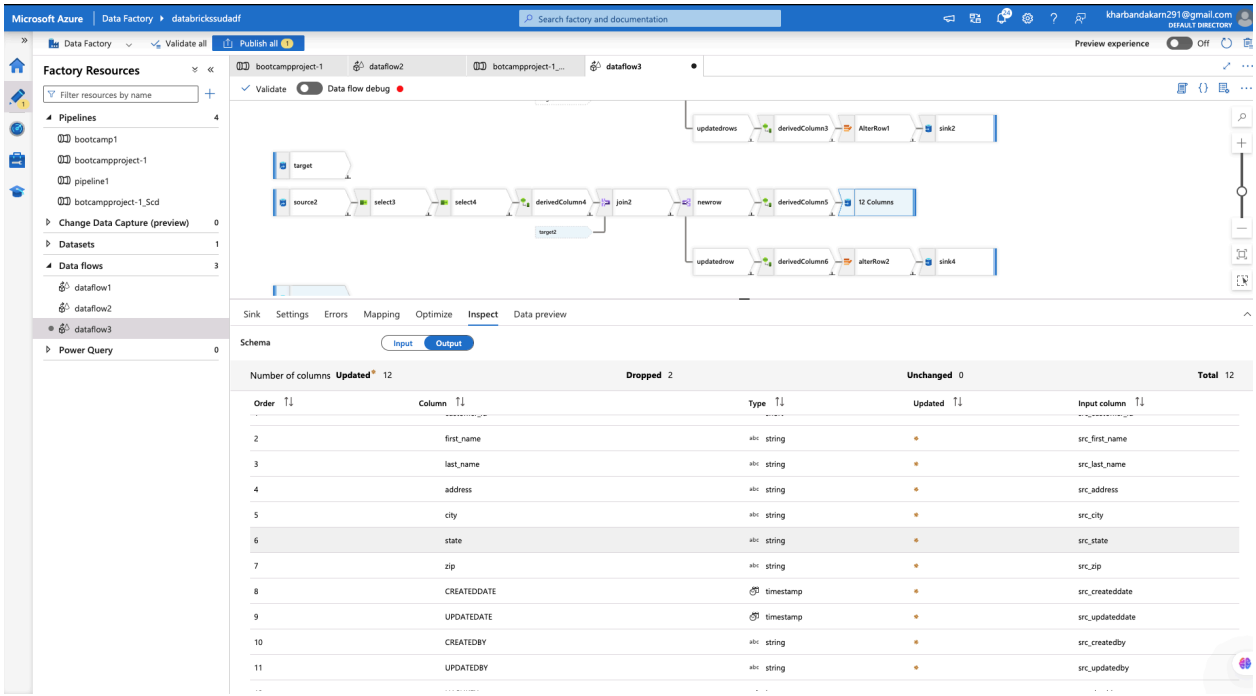
Here will check insert as it update method since this activity will sink new added rows in the SQL database.



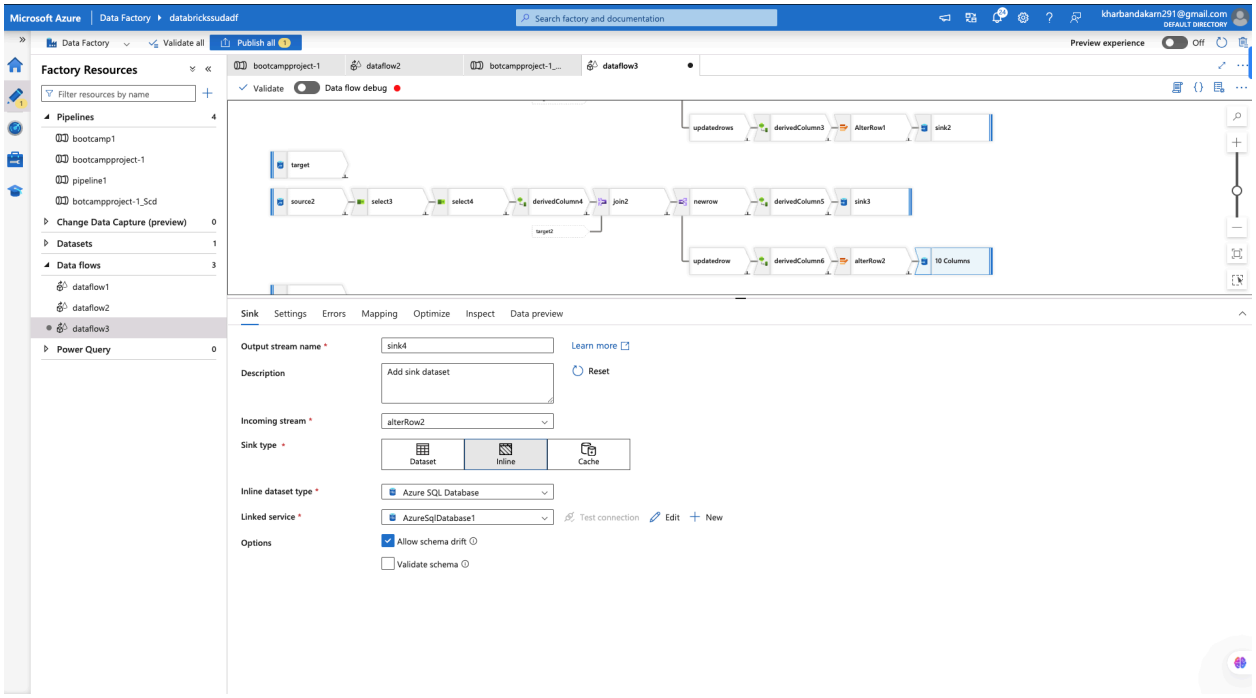
Here is the mapping that we have to new rows stream.



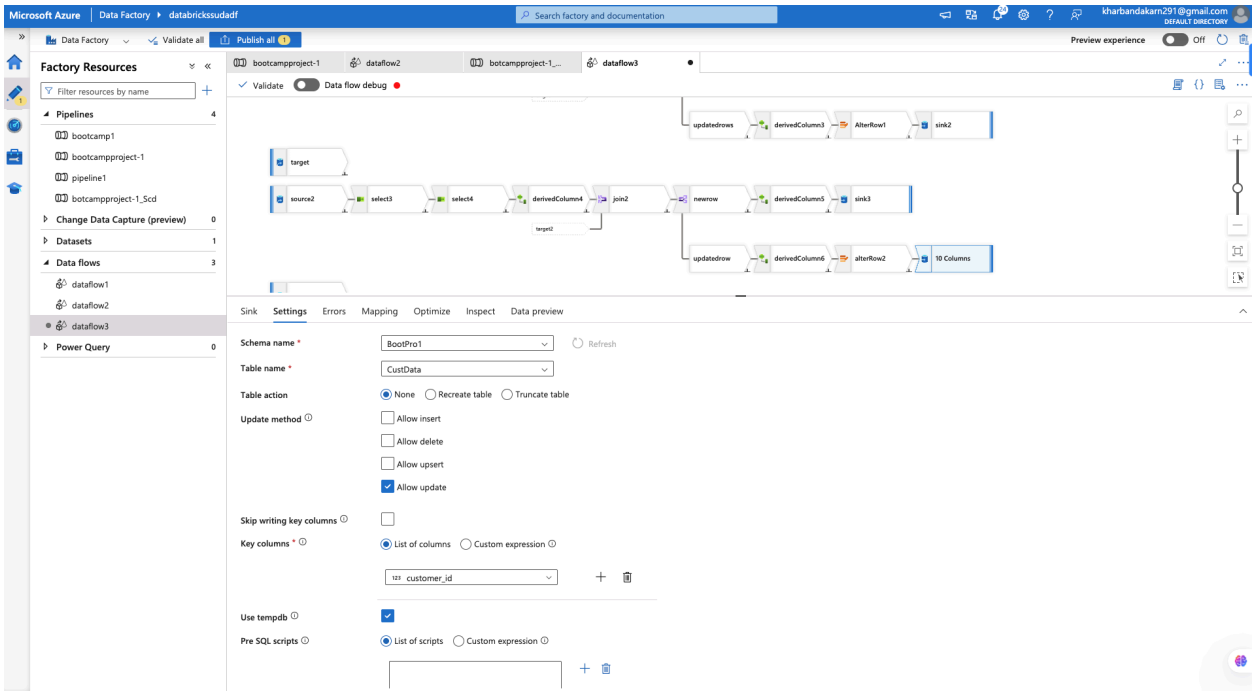
Here is the data review.



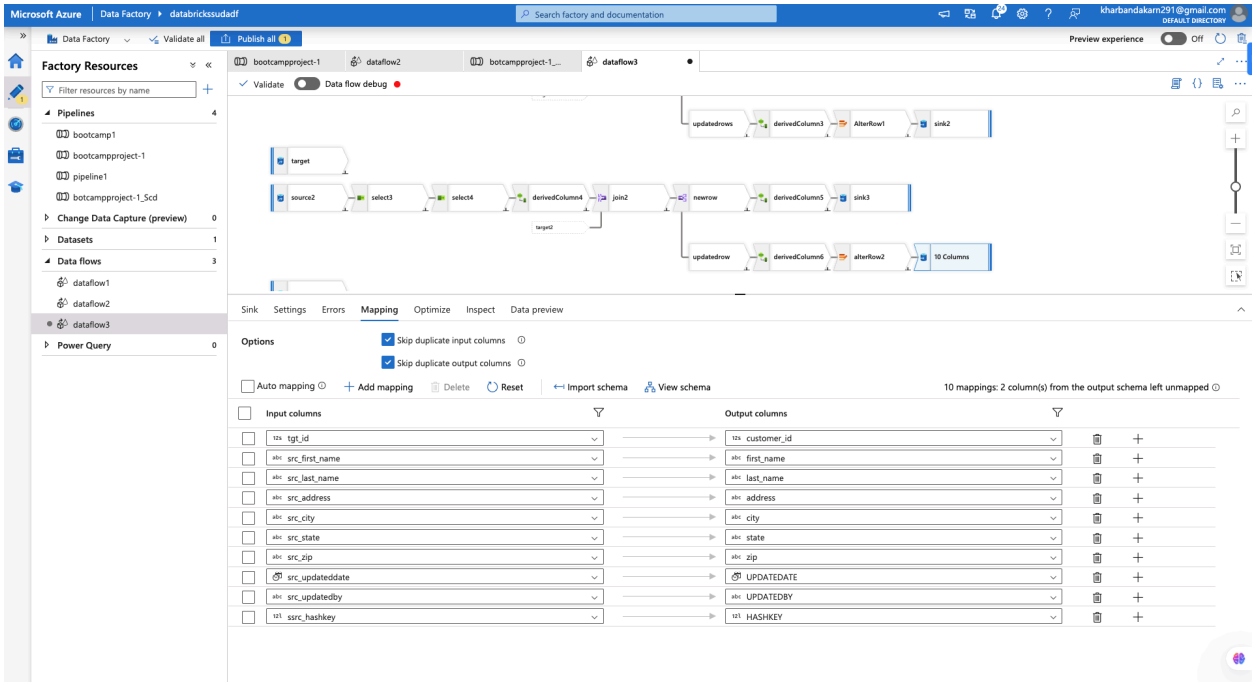
Now we will be adding sink activity where we will be adding new updated row in the sql database.

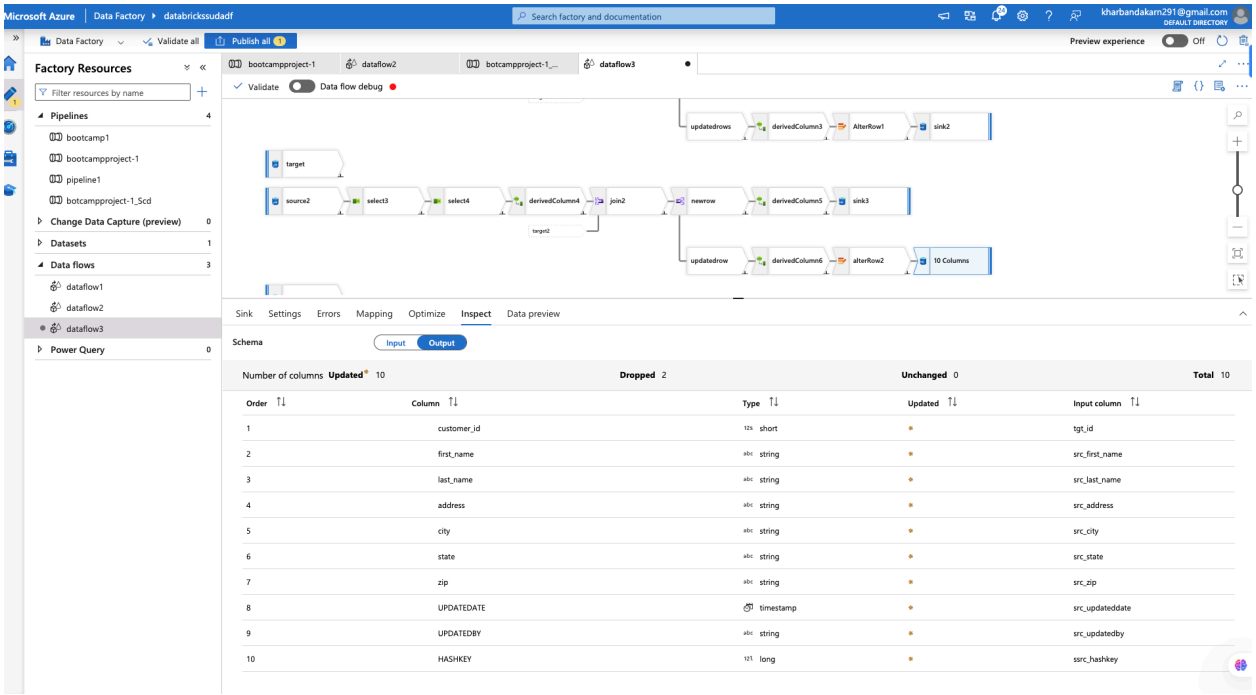


Here we will check update method since we adding updated rows in the database.

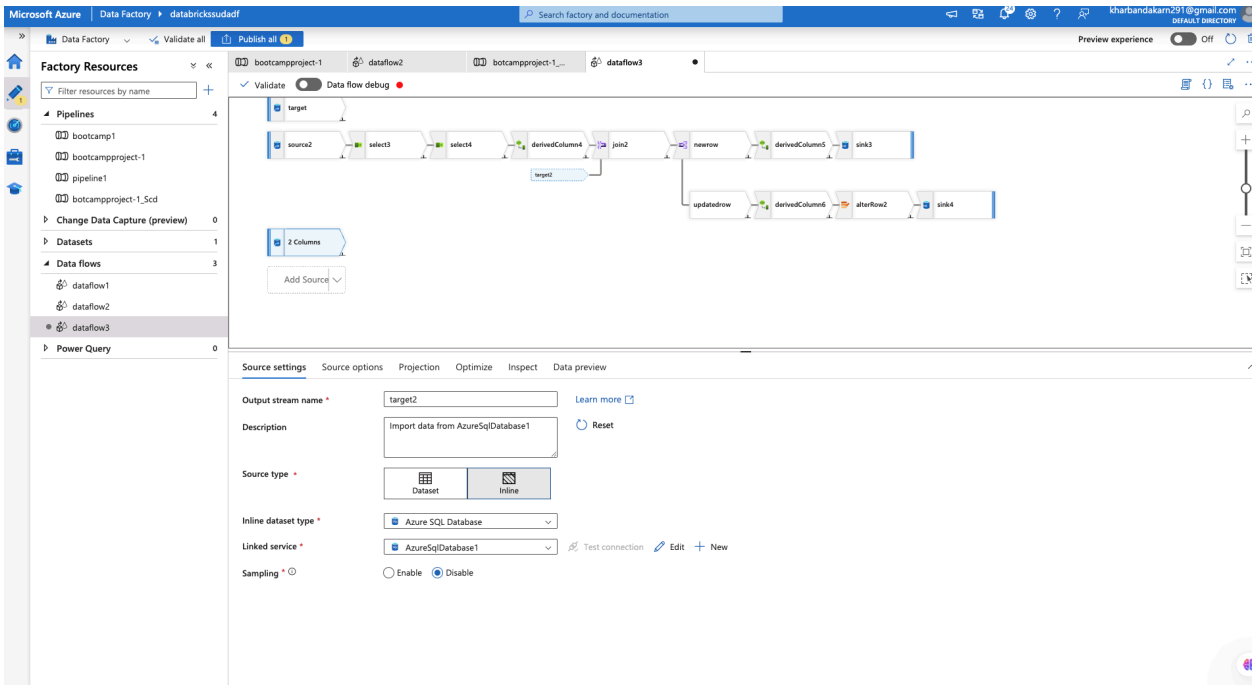


Here is the mapping that we have to done for update row stream.

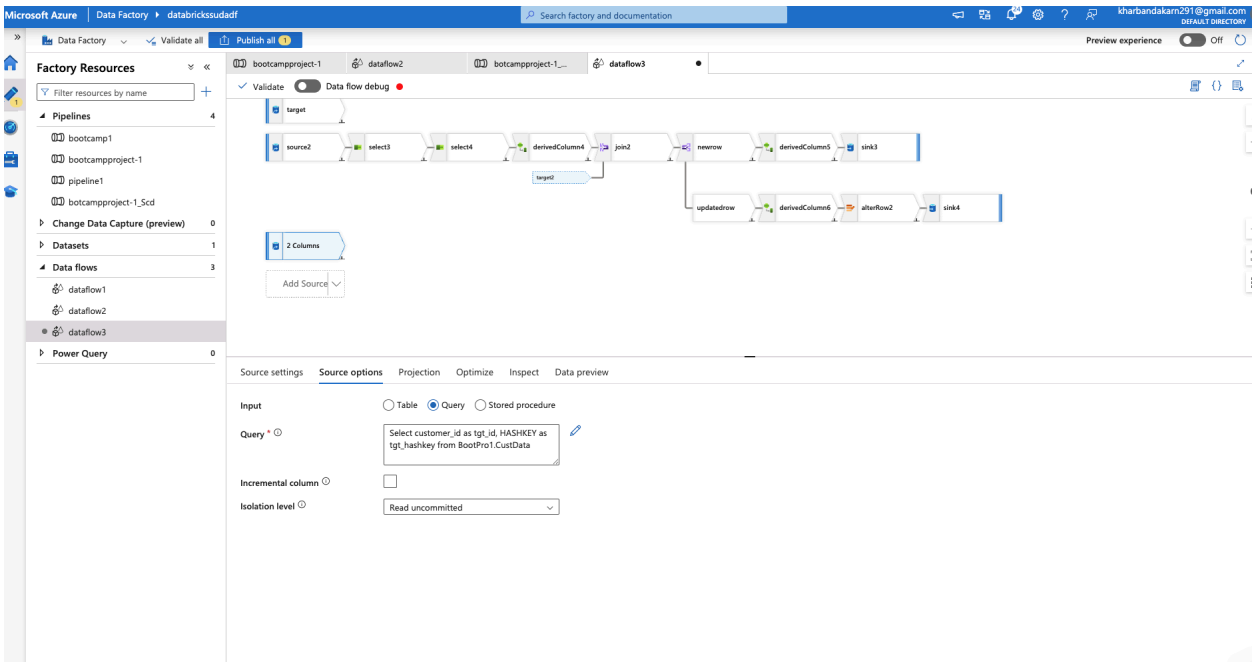




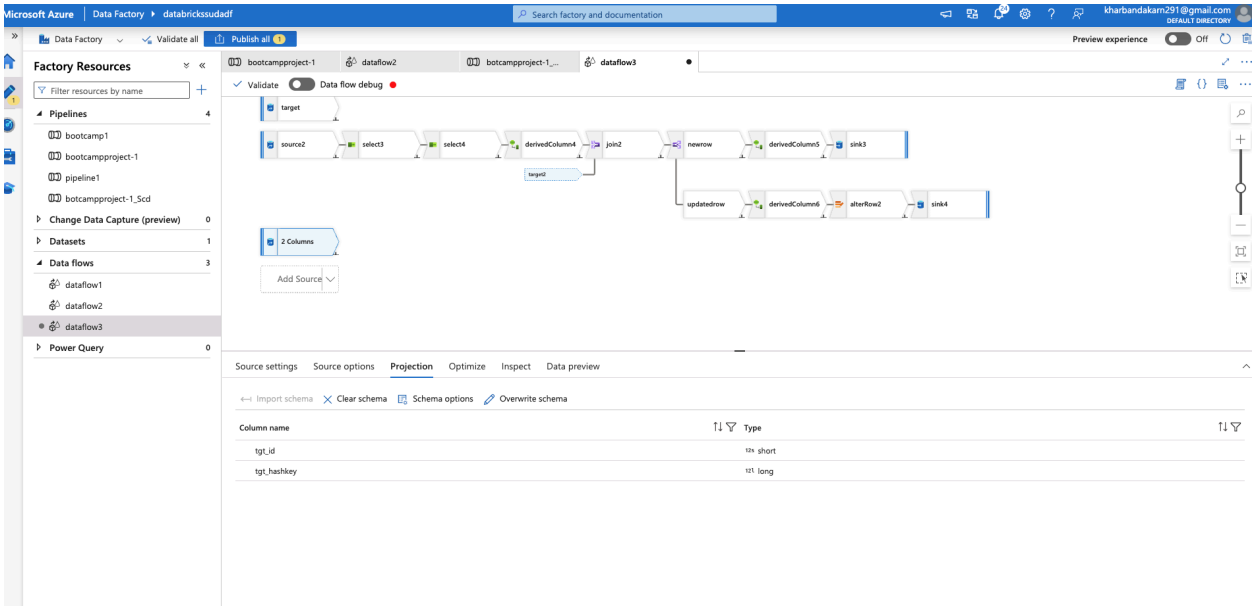
Here is target activity that will loading data from SQL database.



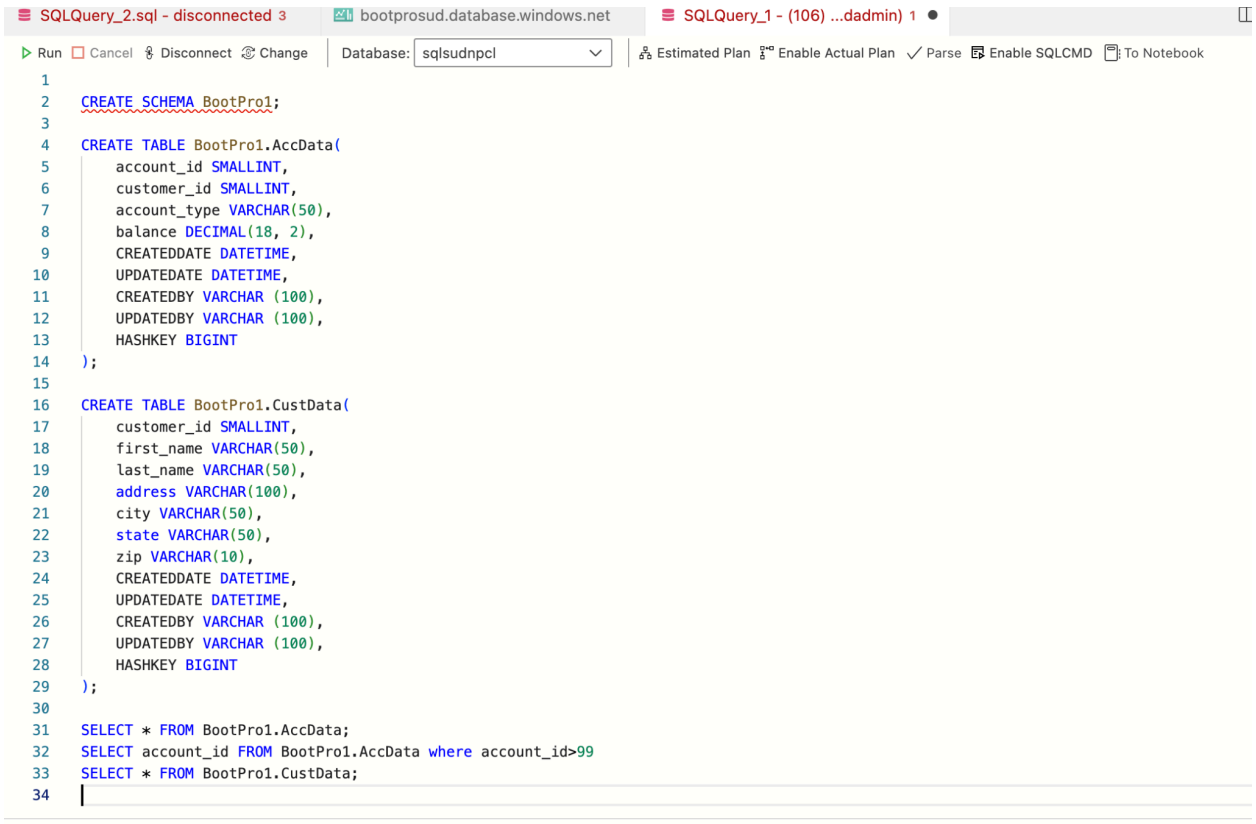
We will using query to load only the primary key and hash key data.



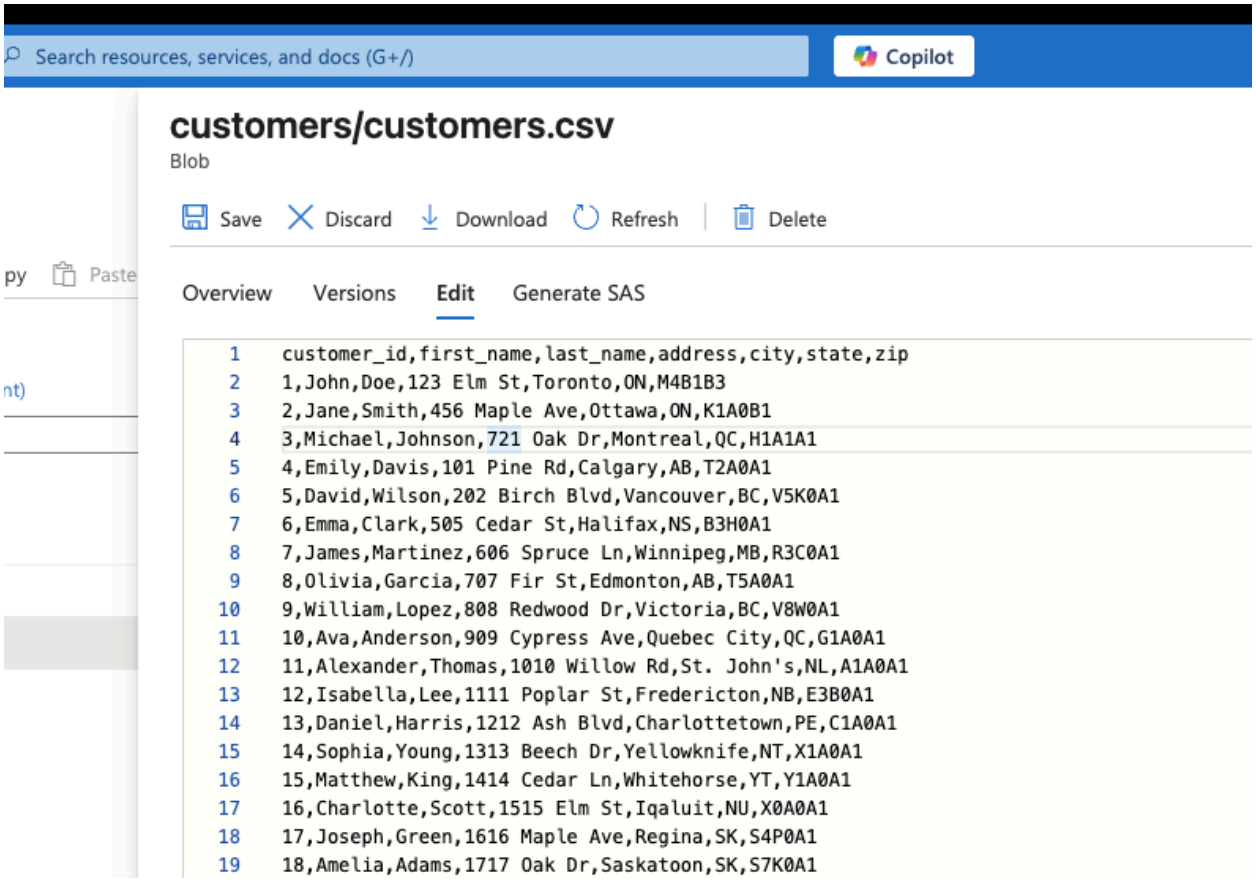
Here is a description of the columns loaded.



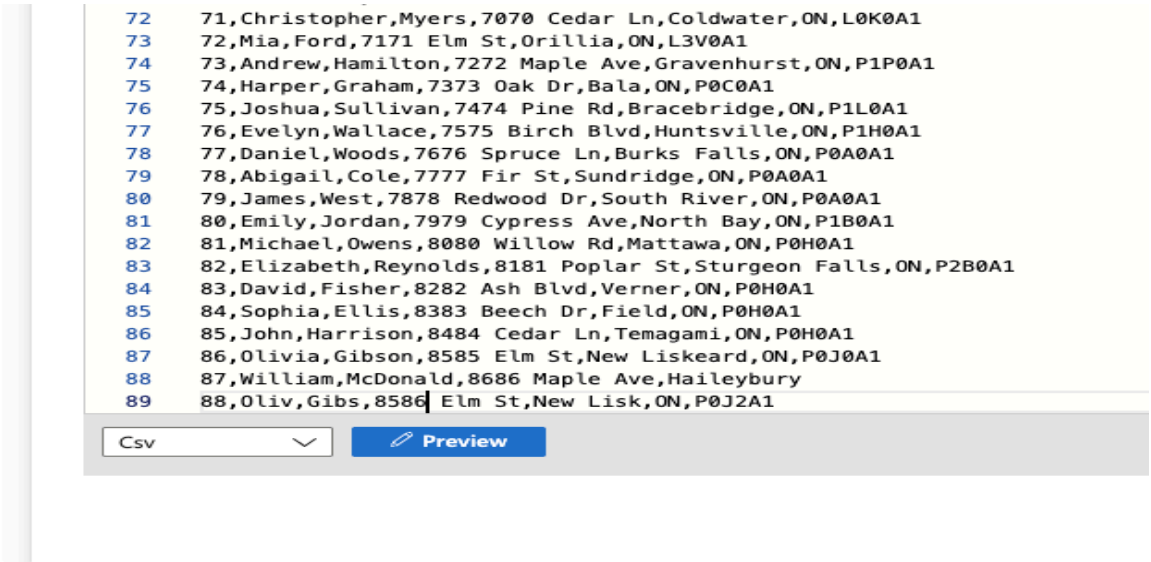
Here is SQL script that we have used for creating scd type -1 table in SQL database.



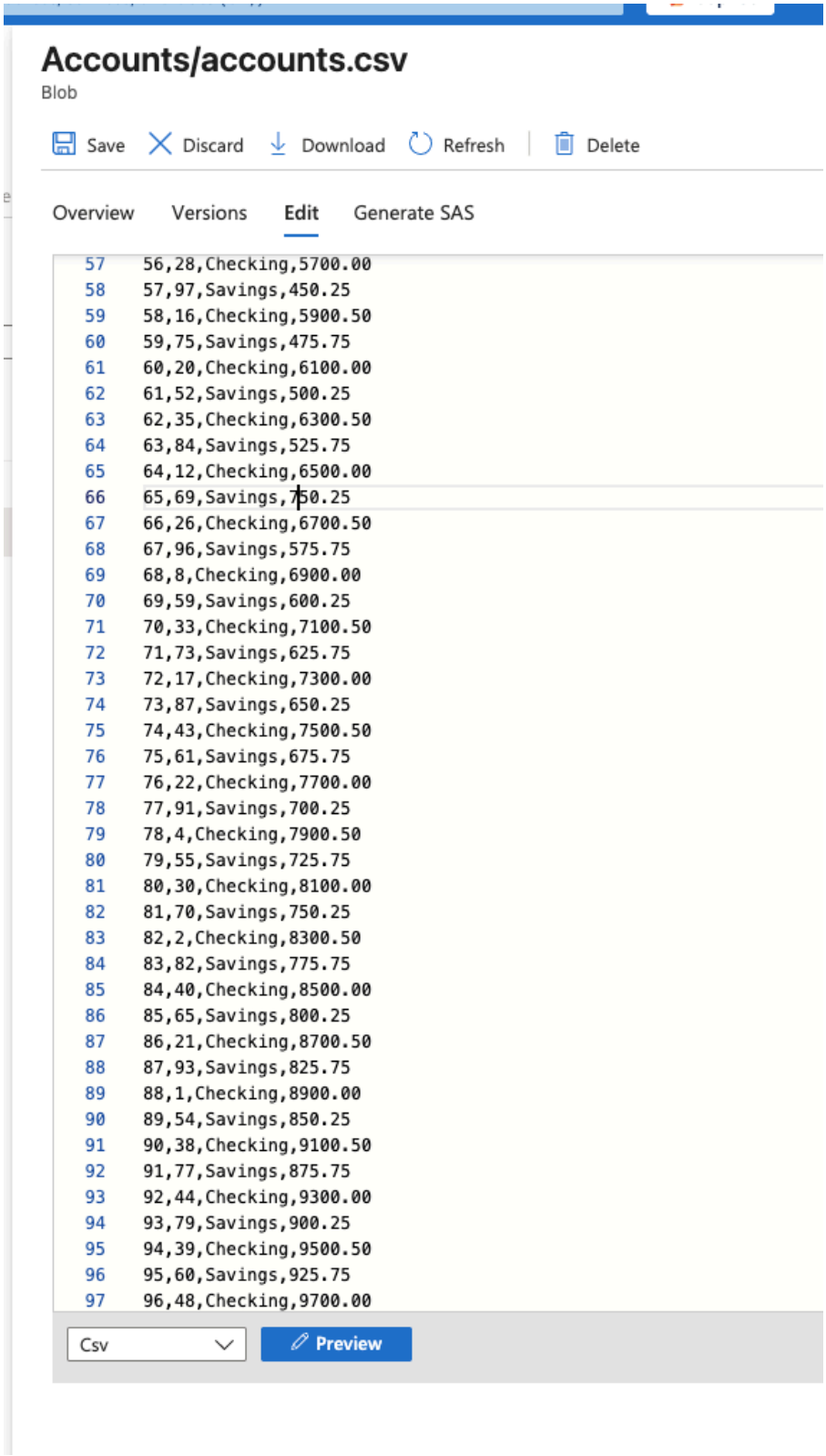
So after running the pipeline for the first time, when all the data has been loaded in the scd type table. We are making changes in the bronze layer where we are updating rows and inserting rows.



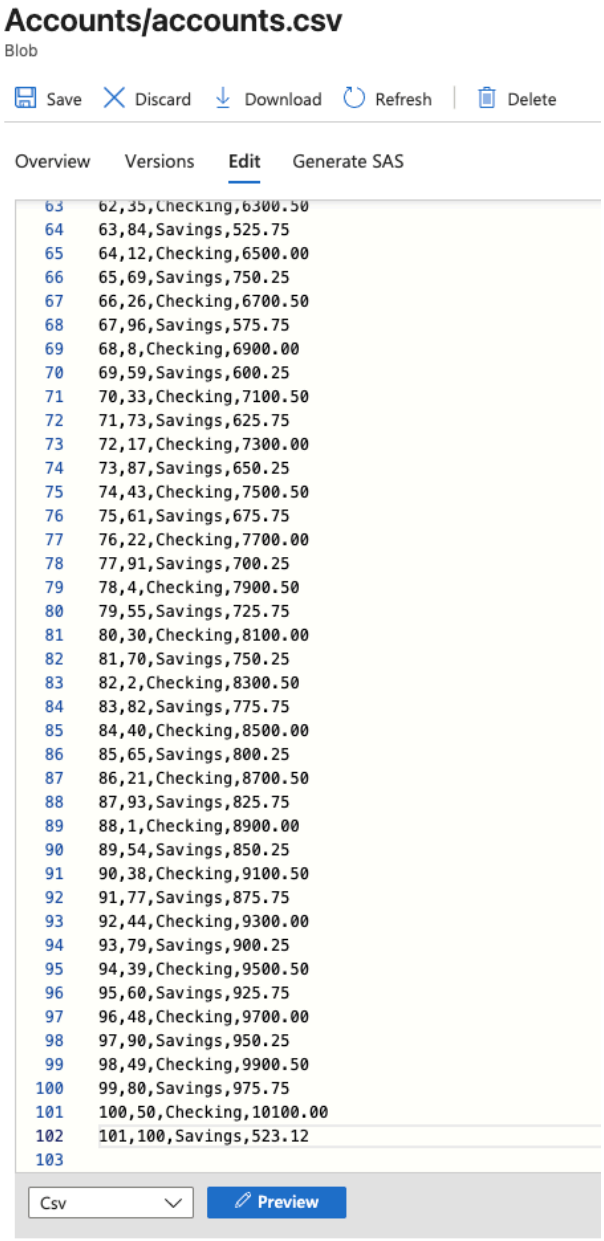
Here we are inserting a new row.



Here are the updated rows in the second CSV file.

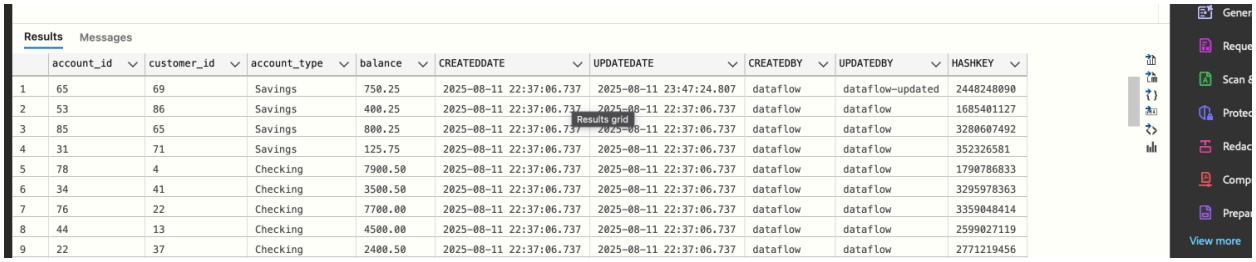


Here we are inserting new rows in the CSV file.



After running the silver layer and gold layer(SCD Type-1) data flow. We are getting the updated rows and newly inserted rows in the SCD-type table.

Updated row

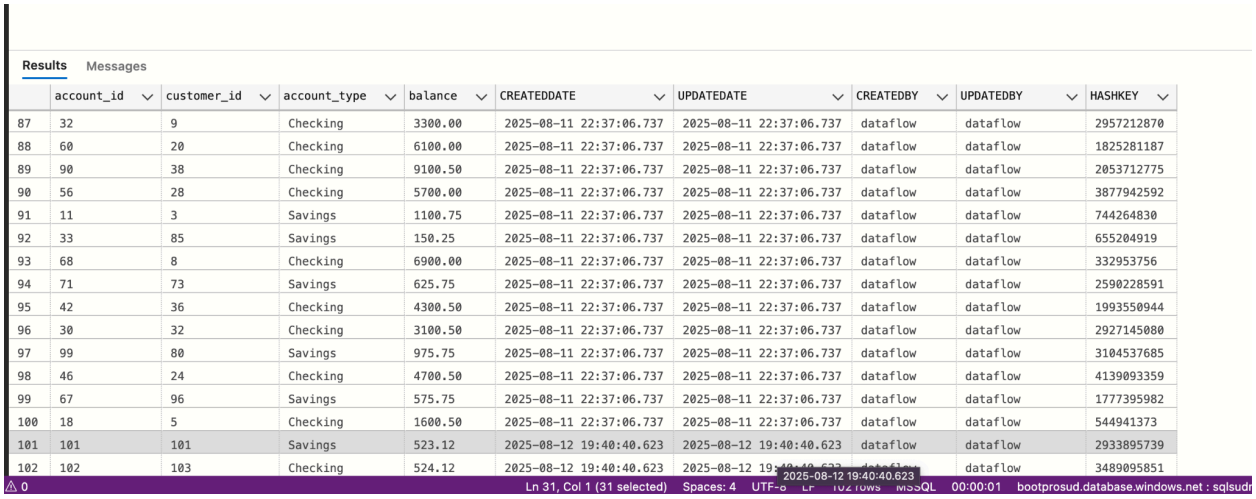


Results | Messages

	account_id	customer_id	account_type	balance	CREATEDDATE	UPDATEDATE	CREATEDBY	UPDATEDBY	HASHKEY
1	65	69	Savings	750.25	2025-08-11 22:37:06.737	2025-08-11 23:47:24.807	dataflow	dataflow-updated	2448248090
2	53	86	Savings	400.25	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	1685401127
3	85	65	Savings	800.25	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	3280607492
4	31	71	Savings	125.75	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	352326581
5	78	4	Checking	7900.50	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	1790786833
6	34	41	Checking	3500.50	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	3295978363
7	76	22	Checking	7700.00	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	3359048414
8	44	13	Checking	4500.00	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	2599027119
9	22	37	Checking	2400.50	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	2771219456

Results grid

Inserted row



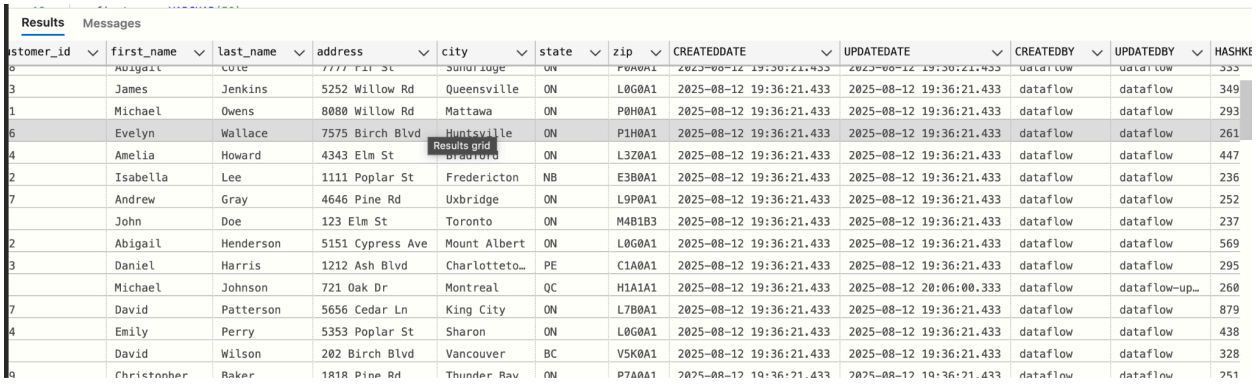
Results | Messages

	account_id	customer_id	account_type	balance	CREATEDDATE	UPDATEDATE	CREATEDBY	UPDATEDBY	HASHKEY
87	32	9	Checking	3300.00	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	2957212870
88	60	20	Checking	6100.00	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	1825281187
89	90	38	Checking	9100.50	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	2053712775
90	56	28	Checking	5700.00	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	3877942592
91	11	3	Savings	1100.75	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	744264830
92	33	85	Savings	150.25	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	655204919
93	68	8	Checking	6900.00	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	332953756
94	71	73	Savings	625.75	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	2590228591
95	42	36	Checking	4300.50	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	1993550944
96	30	32	Checking	3100.50	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	2927145080
97	99	80	Savings	975.75	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	3104537685
98	46	24	Checking	4700.50	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	4139093359
99	67	96	Savings	575.75	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	1777395982
100	18	5	Checking	1600.50	2025-08-11 22:37:06.737	2025-08-11 22:37:06.737	dataflow	dataflow	544941373
101	101	101	Savings	523.12	2025-08-12 19:40:40.623	2025-08-12 19:40:40.623	dataflow	dataflow	2933895739
102	102	103	Checking	524.12	2025-08-12 19:40:40.623	2025-08-12 19:40:40.623	dataflow	dataflow	3489095851

2025-08-12 19:40:40.623

⚠ 0 Ln 31, Col 1 (31 selected) Spaces: 4 UTF-8 LF 102 rows MSSQL 00:00:01 bootprosud.database.windows.net : sqlsudr

Inserted row

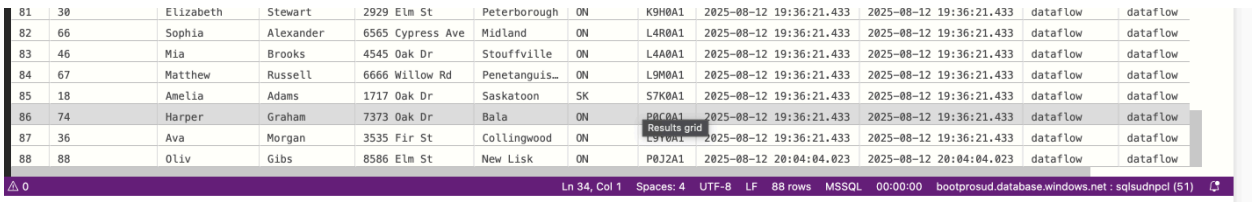


Results | Messages

stomer_id	first_name	last_name	address	city	state	zip	CREATEDDATE	UPDATEDATE	CREATEDBY	UPDATEDBY	HASHKE
3	James	Jenkins	5252 Willow Rd	Queensville	ON	L0G0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	349
1	Michael	Owens	8080 Willow Rd	Mattawa	ON	P0H0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	293
6	Evelyn	Wallace	7575 Birch Blvd	Huntsville	ON	P1H0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	261
4	Amelia	Howard	4343 Elm St	Bradford	ON	L3Z0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	447
2	Isabella	Lee	1111 Poplar St	Fredericton	NB	E3B0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	236
7	Andrew	Gray	4646 Pine Rd	Uxbridge	ON	L9P0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	252
	John	Doe	123 Elm St	Toronto	ON	M4B1B3	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	237
2	Abigail	Henderson	5151 Cypress Ave	Mount Albert	ON	L0G0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	569
3	Daniel	Harris	1212 Ash Blvd	Charlotteto…	PE	C1A0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	295
	Michael	Johnson	721 Oak Dr	Montreal	QC	H1A1A1	2025-08-12 19:36:21.433	2025-08-12 20:06:00.333	dataflow	dataflow-up…	260
7	David	Patterson	5656 Cedar Ln	King City	ON	L7B0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	879
4	Emily	Perry	5353 Poplar St	Sharon	ON	L0G0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	438
	David	Wilson	202 Birch Blvd	Vancouver	BC	V5K0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	328
9	Christopher	Baker	1818 Pine Rd	Thunder Bay	ON	P7A0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow	251

Results grid

Updated row



81	30	Elizabeth	Stewart	2929 Elm St	Peterborough	ON	K9H0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow
82	66	Sophia	Alexander	6565 Cypress Ave	Midland	ON	L4R0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow
83	46	Mia	Brooks	4545 Oak Dr	Stouffville	ON	L4A0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow
84	67	Matthew	Russell	6666 Willow Rd	Penetanguis…	ON	L9M0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow
85	18	Amelia	Adams	1717 Oak Dr	Saskatoon	SK	S7K0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow
86	74	Harper	Graham	7373 Oak Dr	Bala	ON	P0C0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow
87	36	Ava	Morgan	3535 Fir St	Collingwood	ON	L9Y0A1	2025-08-12 19:36:21.433	2025-08-12 19:36:21.433	dataflow	dataflow
88	88	Oliv	Gibs	8586 Elm St	New Lisk	ON	P0J2A1	2025-08-12 20:04:04.023	2025-08-12 20:04:04.023	dataflow	dataflow

Results grid

⚠ 0 Ln 34, Col 1 Spaces: 4 UTF-8 LF 88 rows MSSQL 00:00:00 bootprosud.database.windows.net : sqlsudnpcl (51)