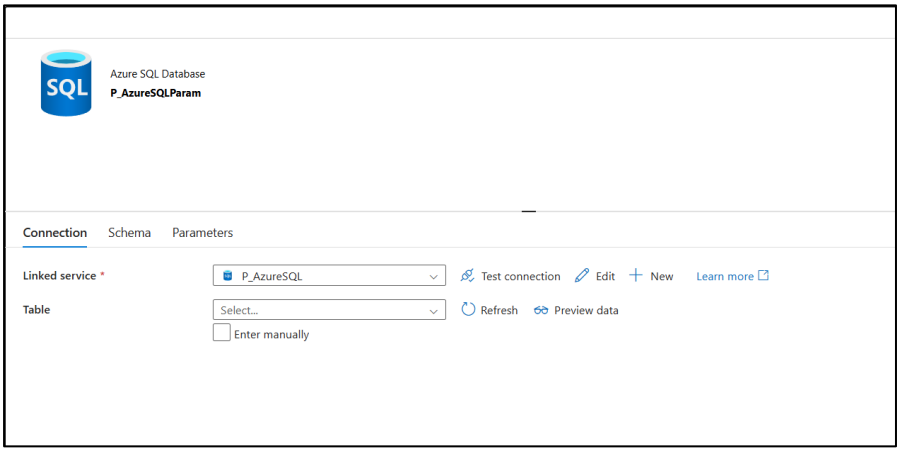
STATEMENT 3

Create a pipeline name 'Foreach_Example2' to copy the below information using just one copy activity. (a) All the product table data where productid > 100 (b) All the customer table data where Customer id>100 and Customer id <1000

We need to use only one **Copy Data** activity, to copy data from product and customer table based on their IDs to ADLS(Azure Data Lake Storage) location in JSON format.

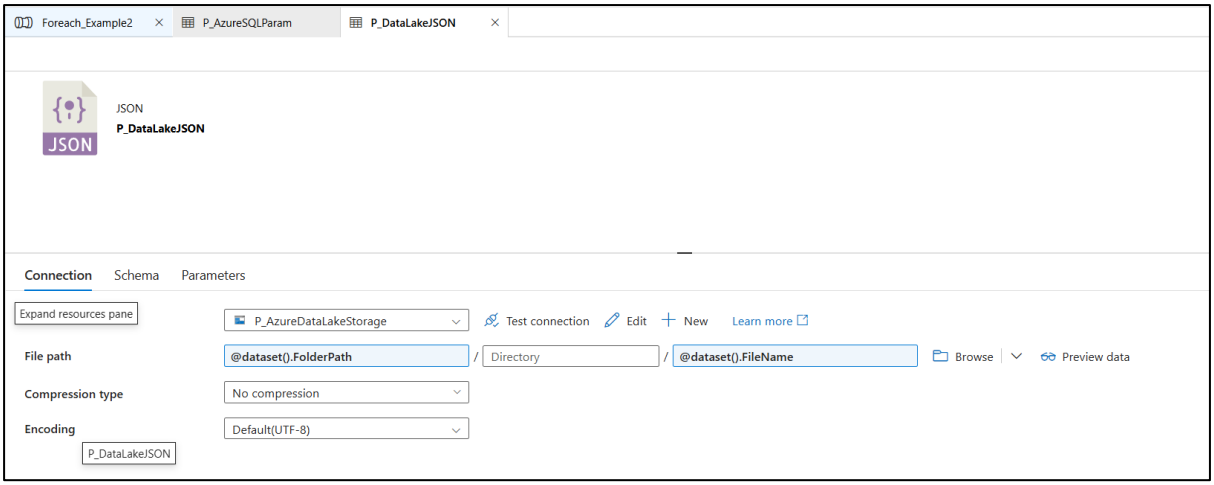
Create Datasets:

1. Create a new dataset referencing to the Azure SQL linked service



Do not provide any table name.

2. Create target dataset to save data as JSON file
The dataset was also used in Statement 1 and Statement 2 pipelines.



Set File System and File Name as @dataset().FolderPath and @dataset().FileName respectively(as shown in the image above).

Set parameters:

3. The Customer dataset has been created in previous statements, the SQL Scripts are present in **customerTable.sql** and **insertCustomers.sql**. (in Azure SQL)
4. Create Product table and insert data (in Azure SQL)

```
CREATE TABLE Product (
    ProductID INT PRIMARY KEY,
    ProductName VARCHAR(100),
    Category VARCHAR(50),
    Price DECIMAL(10,2),
    CreatedDate DATE
);

-- Insert sample data
INSERT INTO Product VALUES
(1, 'Notebook', 'Stationery', 3.50, '2025-07-01'),
(55, 'Pen', 'Stationery', 1.20, '2025-07-02'),
(101, 'Laptop', 'Electronics', 800.00, '2025-07-03'),
(102, 'Smartphone', 'Electronics', 600.00, '2025-07-04'),
(205, 'Office Chair', 'Furniture', 120.00, '2025-07-05'),
(307, 'Standing Desk', 'Furniture', 300.00, '2025-07-06');
```

The script is present in **insertProduct.sql**

Step-Wise Guidelines:

1. Create a pipeline named: Foreach_Example2
2. Create pipeline parameter

   Name: Queries

   Type: Array

   Value: [

   {

   "tableName":"product",

   "query":"SELECT * FROM Product WHERE ProductID > 100"

   },

   {"tableName":"customer",

   "query":"SELECT * FROM customer WHERE customer_id > 100 AND customer_id < 1000"

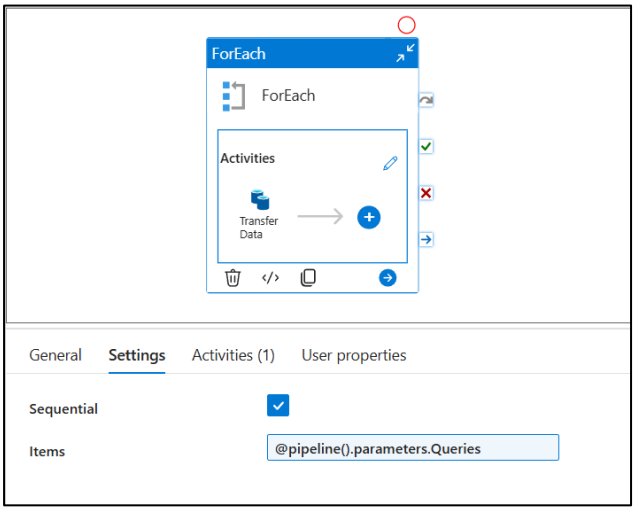
   }

   ]

   The queries can be viewed in **queries.sql** script

   The value contains the SQL table name and their corresponding SQL query to be performed.

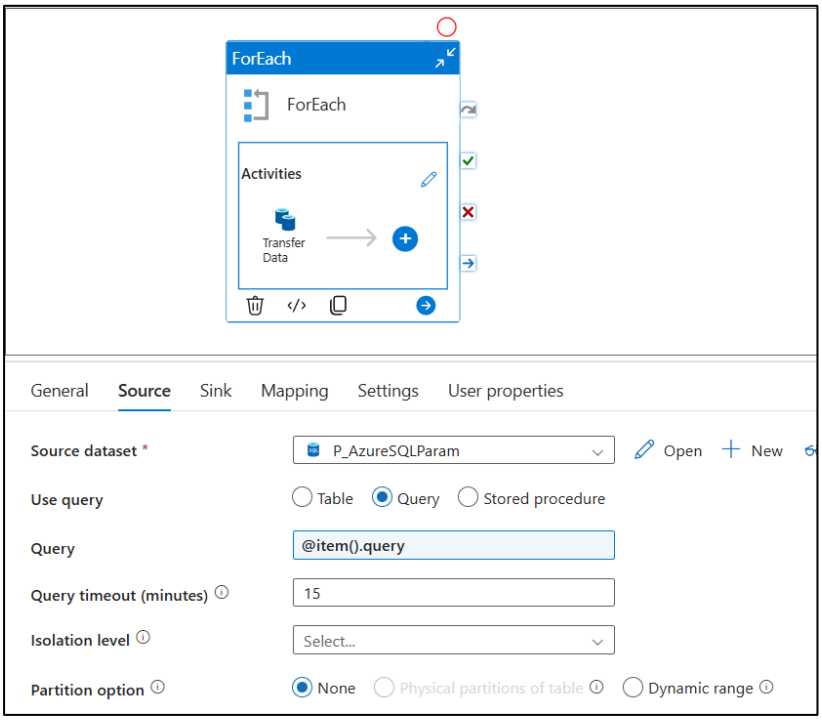
3. Insert a **ForEach** activity



   Specify **Items:** @pipeline().parameters.Queries

   The **"Items"** property in a **ForEach** activity defines the **collection of items** that the activity will iterate over. It acts as the input array for the loop, and ADF will execute the inner activities **once for each item** in that collection.

4. Insert **Copy Data** within ForEach activity

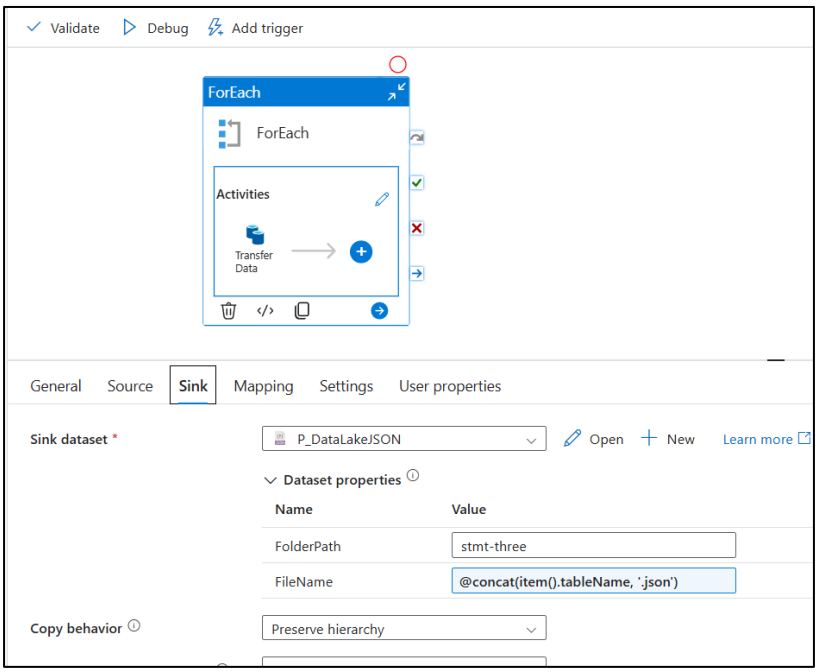


Where,

Source Dataset is the dataset we created(see Page 1)

Query: @item().query

In each iteration we are accessing the query from the parameter values and running on source dataset



Where,

Sink Dataset is the JSON format dataset we created on Page 1

Provide parameter values:

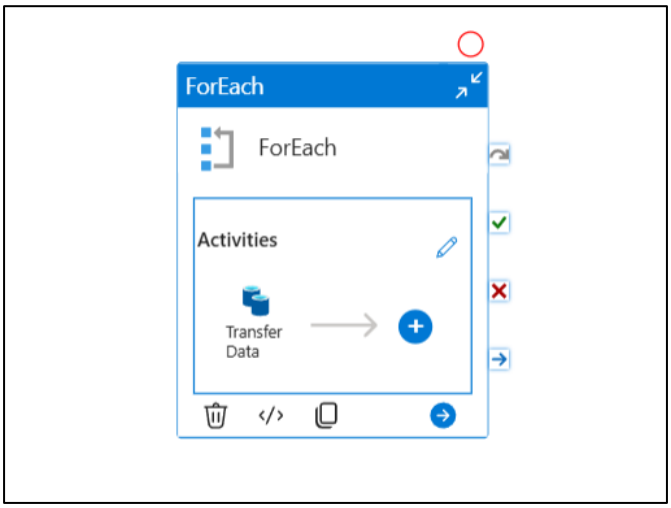
FolderPath: stmt-three (Directory in which data will be copied)

FileName: @concat(item().tableName, '.json')

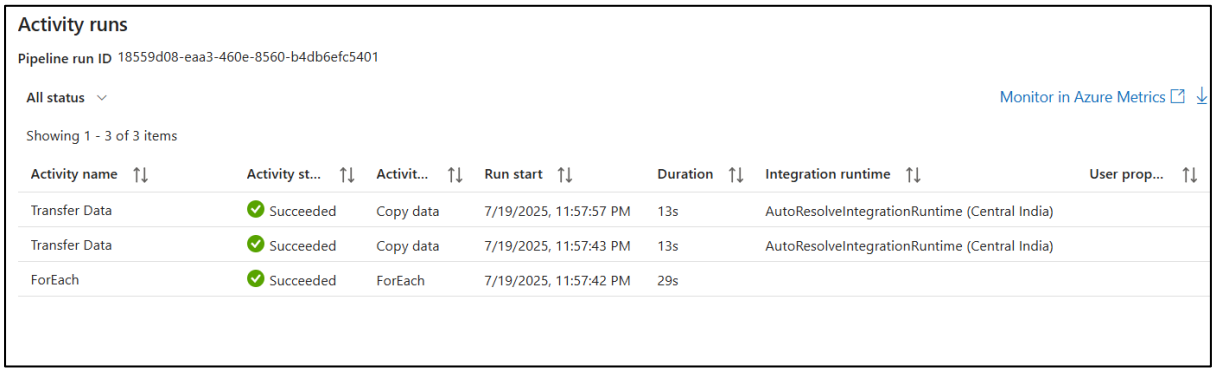
We are dynamically creating file name based on the tableName present in the Queries parameter

Use **preserve hierarchy** as copy behaviour and file pattern must be **Array of objects**.

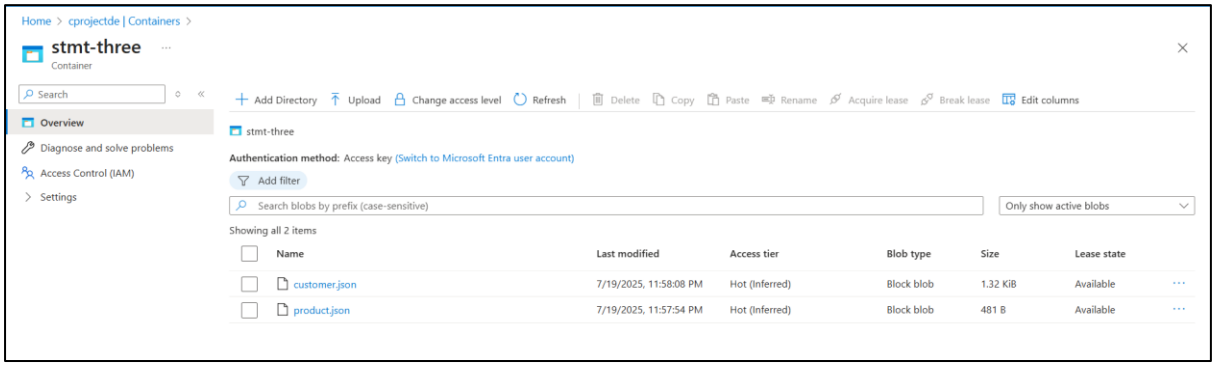
Final Pipeline:



Pipeline Execution:

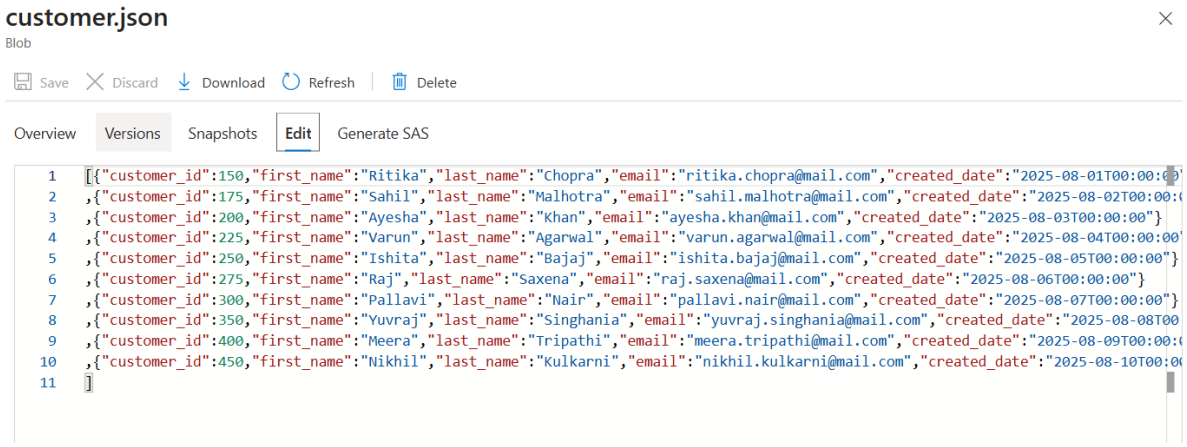


Data Copied as JSON:



Data is transferred to their respective JSON files in stmt-three directory present in cprojectde storage account.

customer.json



Product.json