STATEMENT 2

Every time the pipeline runs it will overwrite the data. Ensure that all the data goes into Folder like (Customer/Year/Month/Day) to avoid any overwriting. Please rewrite the pipeline.

In the previous statement/problem we built a pipeline, which copied data from Customer table, Azure SQL into ADLS(Azure Data Lake Storage) storage in JSON format. However, each time the operation is performed, the data is overwritten in the file. Therefore, now, we have to store the data in folder hierarchy: Customer/Year/Month/Day, where, within the 'Custoemr' folder we have to dynamically create the Year, Month and Day folder corresponding to the time at which the copy operation is executed.

For eg, If we are performing the operation on 20th July, 2025, the data will be saved in **Customer/2025/July/20**.

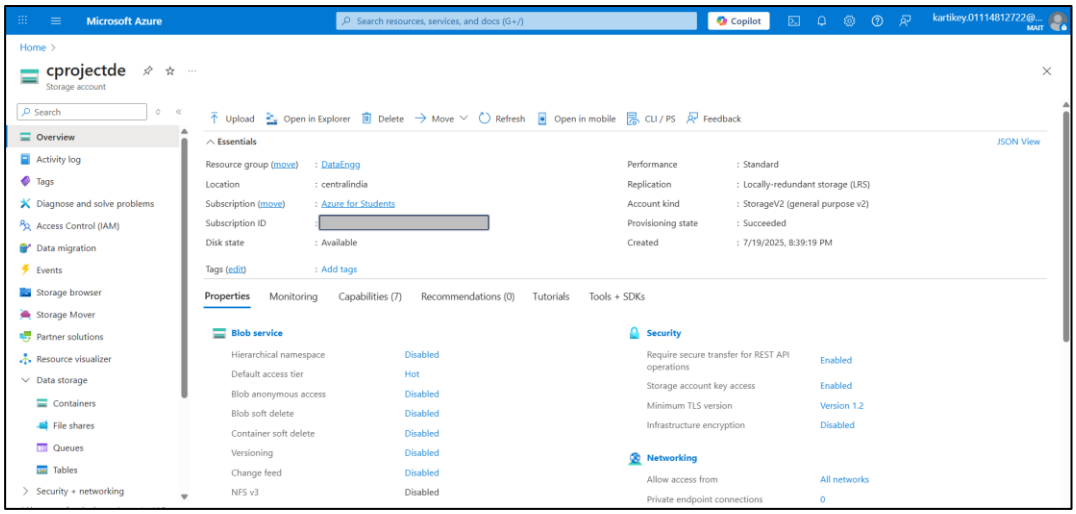
Most of the operations and activities to be performed are similar to the previous statement, we just need to create dynamic subdirectories according to the time of pipeline execution.

Prerequisites:

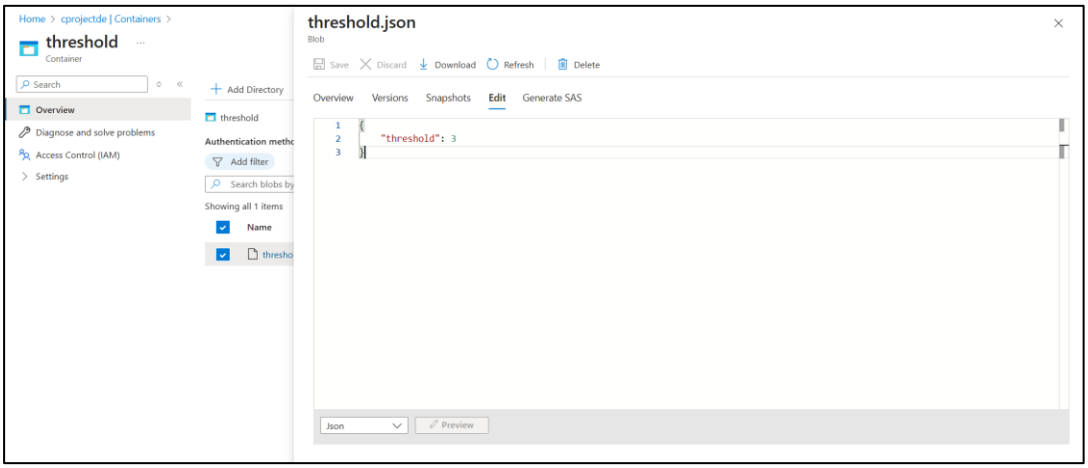
1. Threshold file in ADLS location

   For storing the file in ADLS location, we need to create a **storage account** in Azure. Provide the

   Subscription
   Resource Group
   Storage Account Name
   Set **Redundancy** accordingly.(account used in this project has Locally-Redundant Storage)



Create a threshold file in **Containers**



The image shows **threshold.json** file in threshold directory of **cprojectde** storage account, with a threshold value of 3.

2. Customer table in Azure SQL

Creating table:

```
CREATE TABLE customer (
    customer_id INT PRIMARY KEY,
    first_name VARCHAR(50),
    last_name VARCHAR(50),
    email VARCHAR(100),
    created_date DATETIME
);
```

Inserting values:

```
INSERT INTO customer VALUES

(1, 'Rahul', 'Gupta', 'rahul@mail.com', '2025-07-11'),

(2, 'Prerna', 'Aggarwal', 'prerna@mail.com', '2025-07-12'),

….

(1009, 'Manish', 'Gupta', 'manish.gupta@mail.in', '2025-08-25'),

(1010, 'Divya', 'Singh', 'divya.singh@mail.in', '2025-08-26'),
```

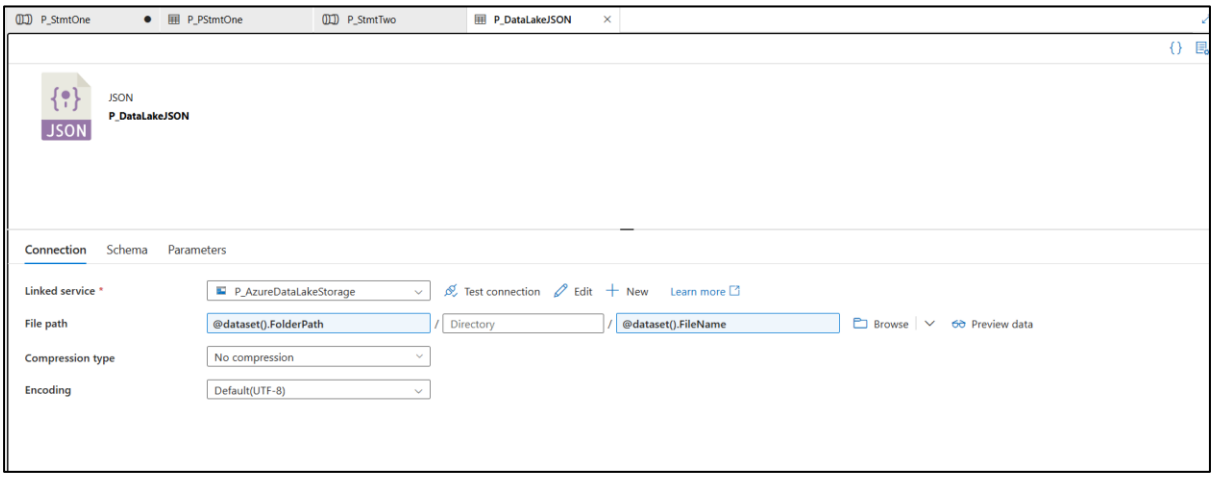
The full scripts are present in **insertCustomers.sql** and **customerTable.sql**

Total records in Customer table: **71**

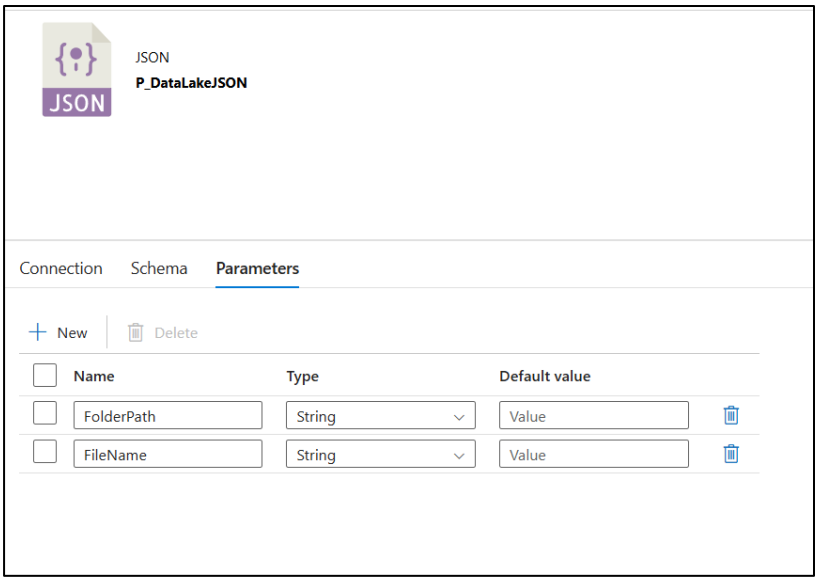
Add the files and table created as datasets in your Azure Data Factory

NOTE: If no data factory has been created yet, see Page 5

1. **Threshold File Dataset and target file dataset**
   Create **Azure Data Lake Storage Gen2** dataset.
   The format of the dataset should be **JSON**



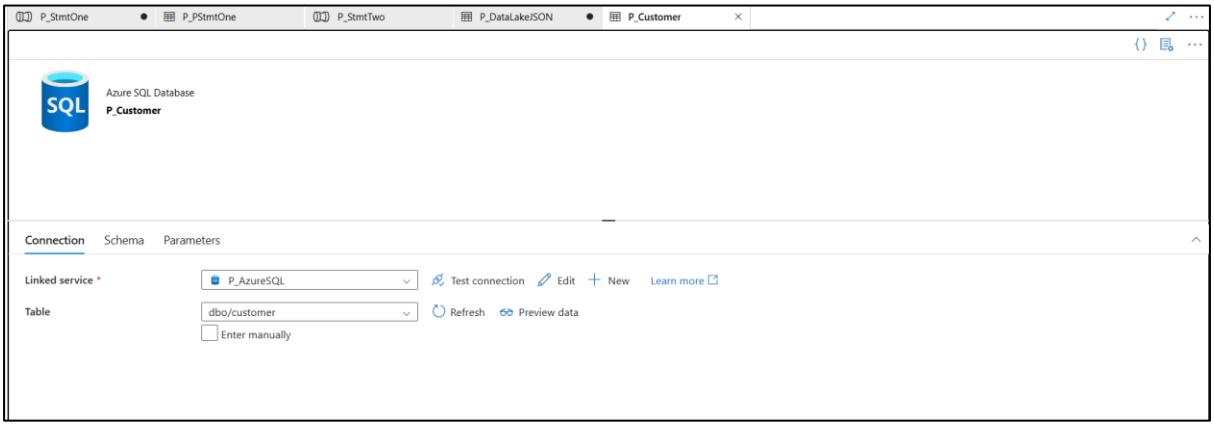
   Create two parameters:



   Populate the **File Path:**
   File System: @dataset().FolderPath
   File Name: @dataset().FileName

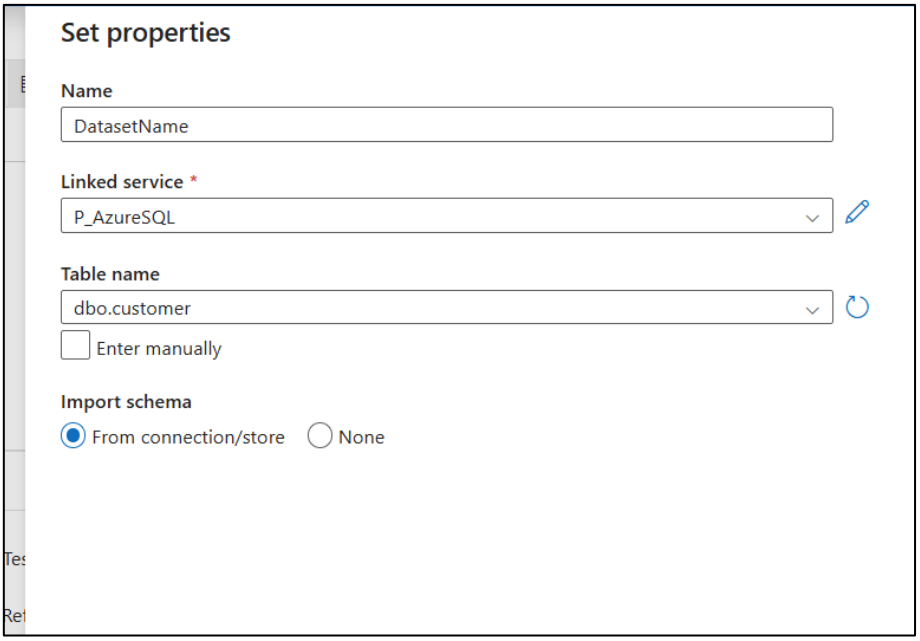
2. SQL Table Dataset

Create Azure SQL dataset

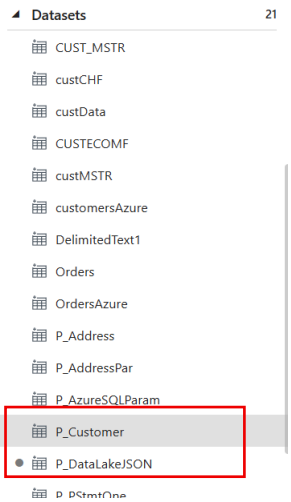


Provide:

Dataset Name
Linked Service
Table Name



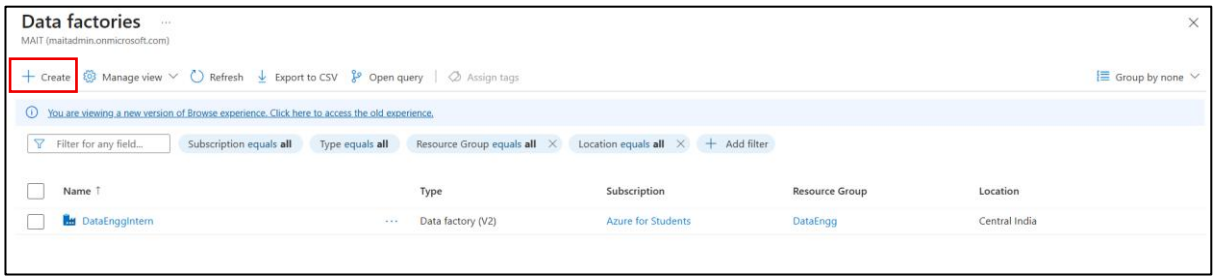
Final

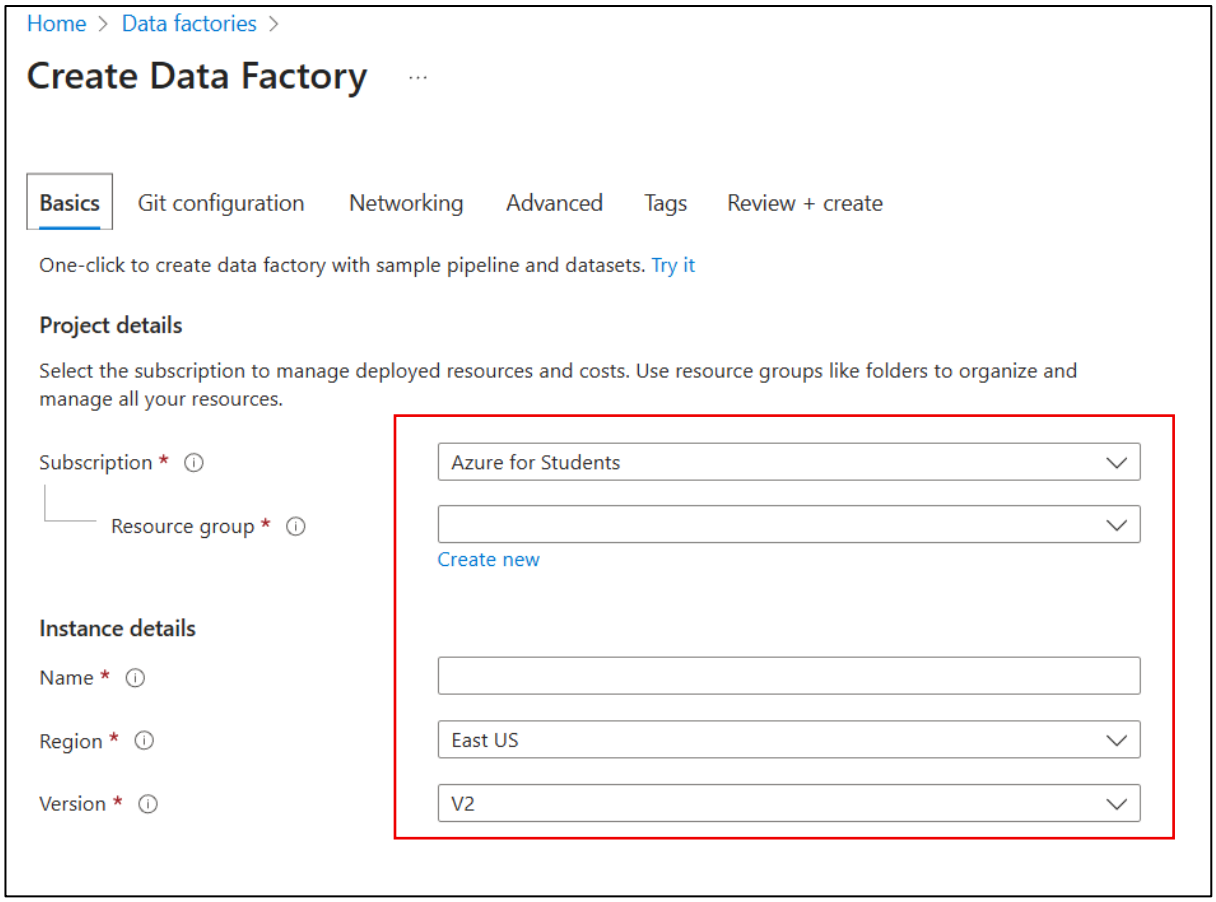


Steps taken to achieve the required goal:

We have to create a pipeline in Azure Data Factory.

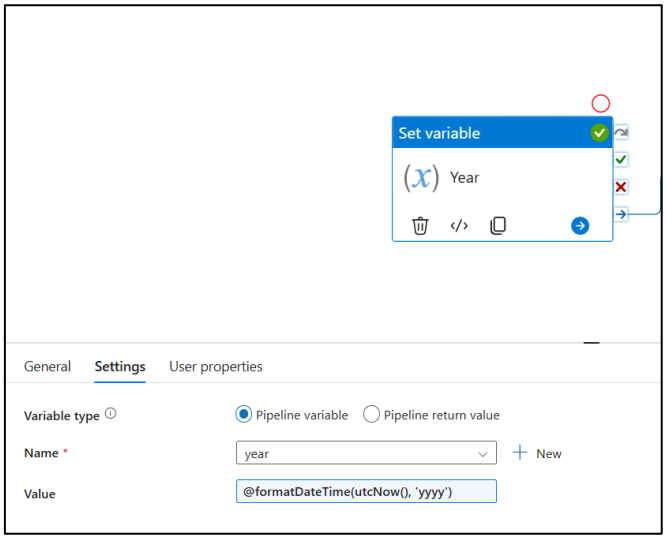
NOTE: Create an Azure Data Factory if not already created.



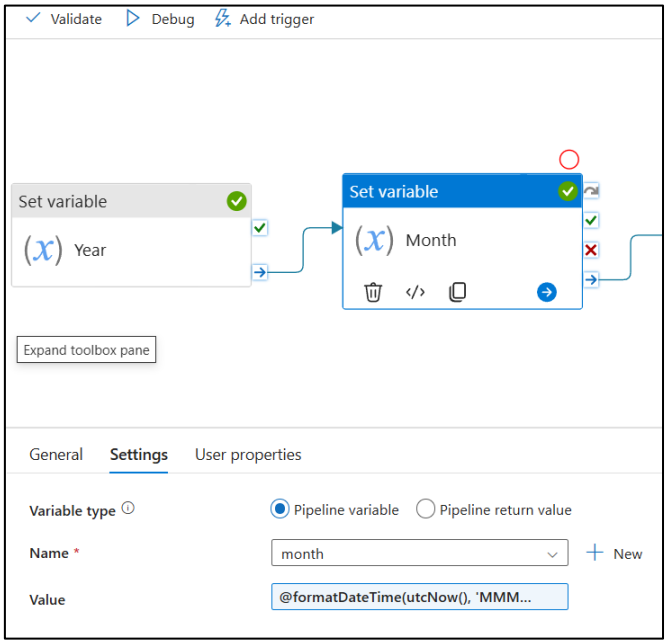


After filling the details, click, **Review + Create**

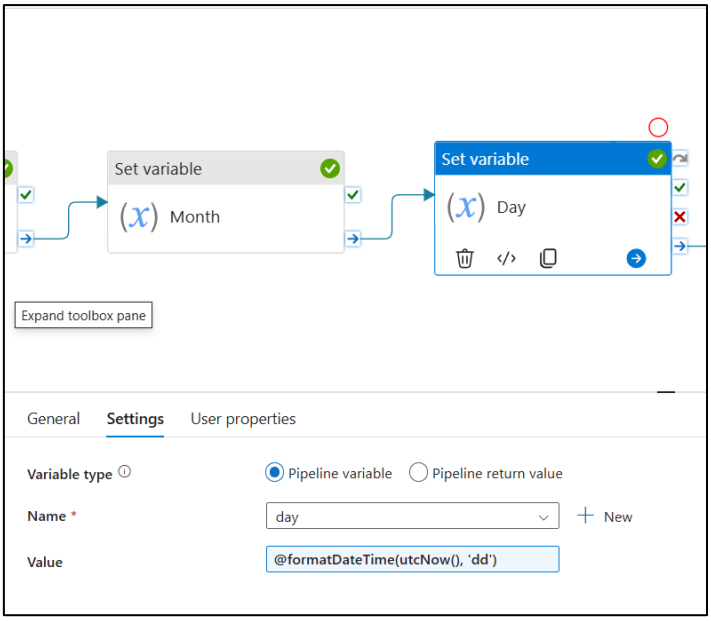
1. Create new pipeline, name it accordingly.
2. We will create three variables, that store the year, month and date of the pipeline execution. Activity to be used is **Set Variable**



Variable: year
Value: @formatDateTime(utcNow(), 'yyyy')

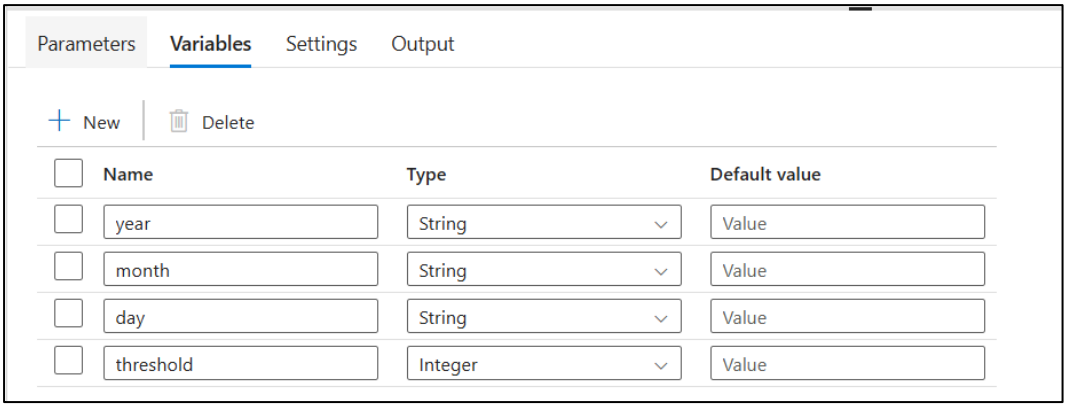


Variable: month
Value: @formatDateTime(utcNow(), 'MMMM')

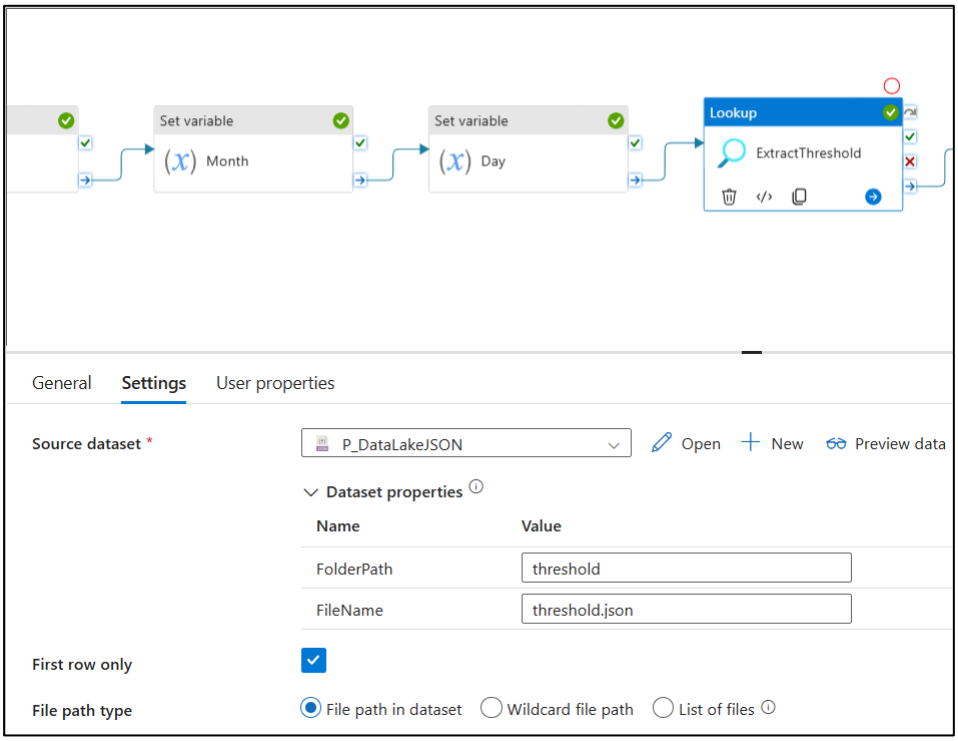


Variable: day
Value: @formatDateTime(utcNow(), ‘dd’)

Pipeline Variables Initialised as:



3. Insert **Lookup** Activity



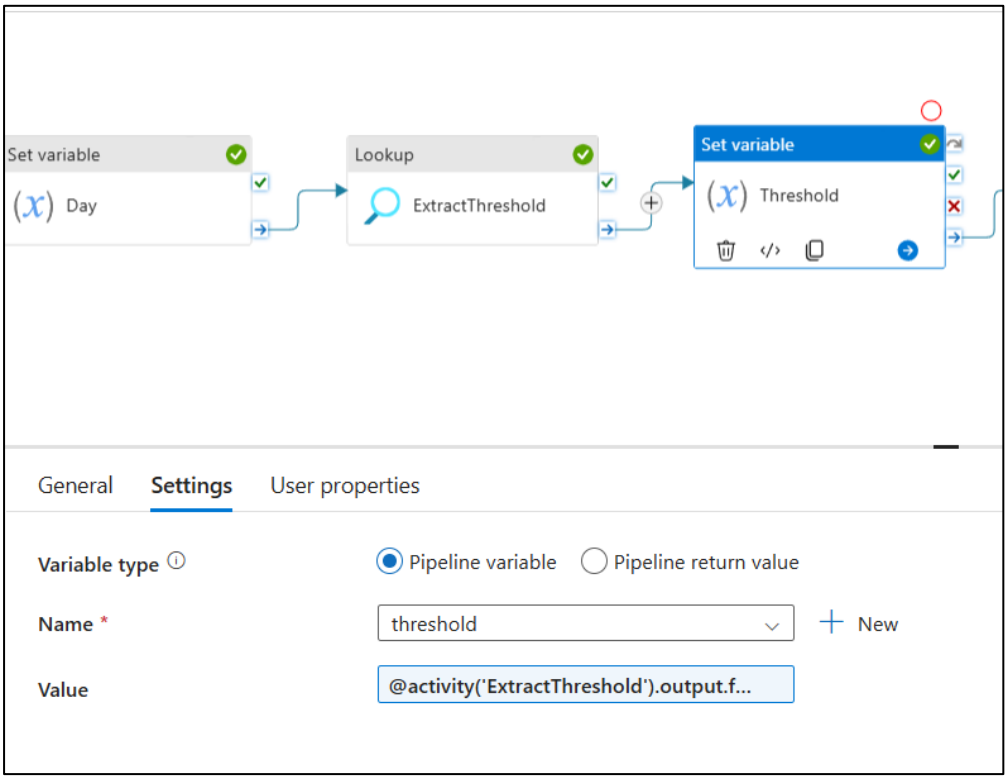
The name of the **Lookup** activity as shown above is ExtractThreshold.
The source dataset is the dataset, where threshold.json is present.
We can extract the file using the parameter we created during dataset creation(Page 3)

The parameters

FolderPath: Name of the direcory in which json file is present(can be left blank if file is not in directory)

FileName: Name of the file(eg. threshold.json)

4. Insert **Set Variable** activity



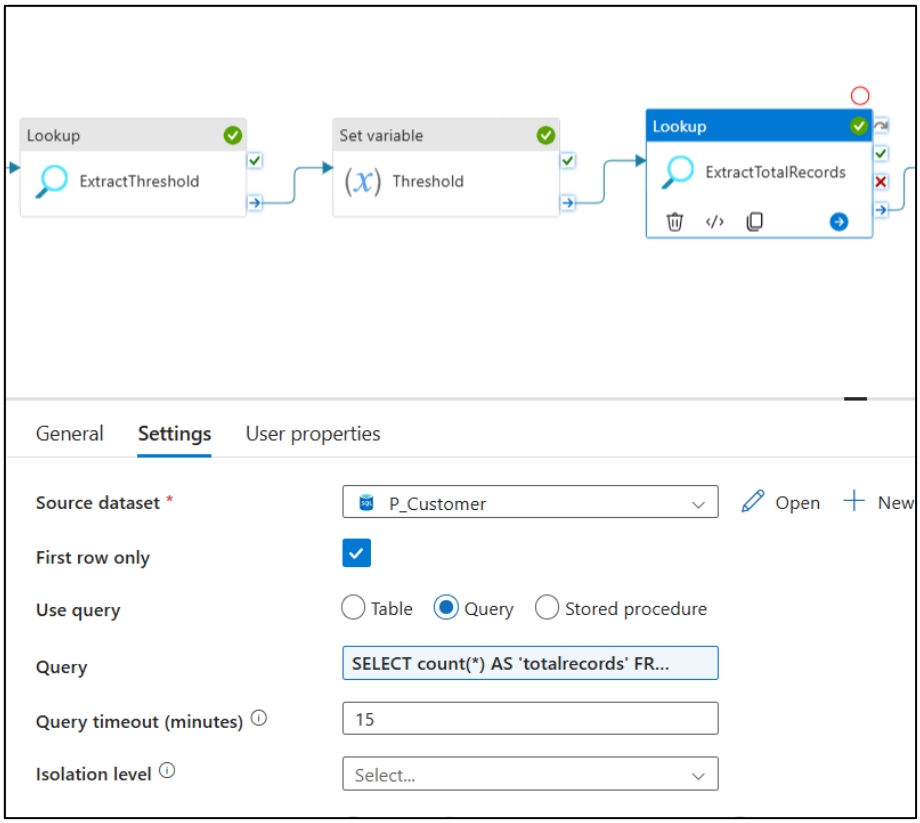
Value of variable: @activity('ExtractThreshold').output.firstRow.threshold
This variable extracts the threshold value from the output of Lookup Activity. The value is according to the file format, in our case the file is:

```
{
        "threshold": 3
}
```

Where, the first row contains the **threshold** parameter, we extract this using firstRow.threshold (see **Value**)

For this, initialise variable 'threshold' in the pipeline of type integer. (as shown on Page 7)

5. Insert **Lookup** Activity



This Lookup activity extracts the total number of records present in the Customer table in Azure SQL.

Where,

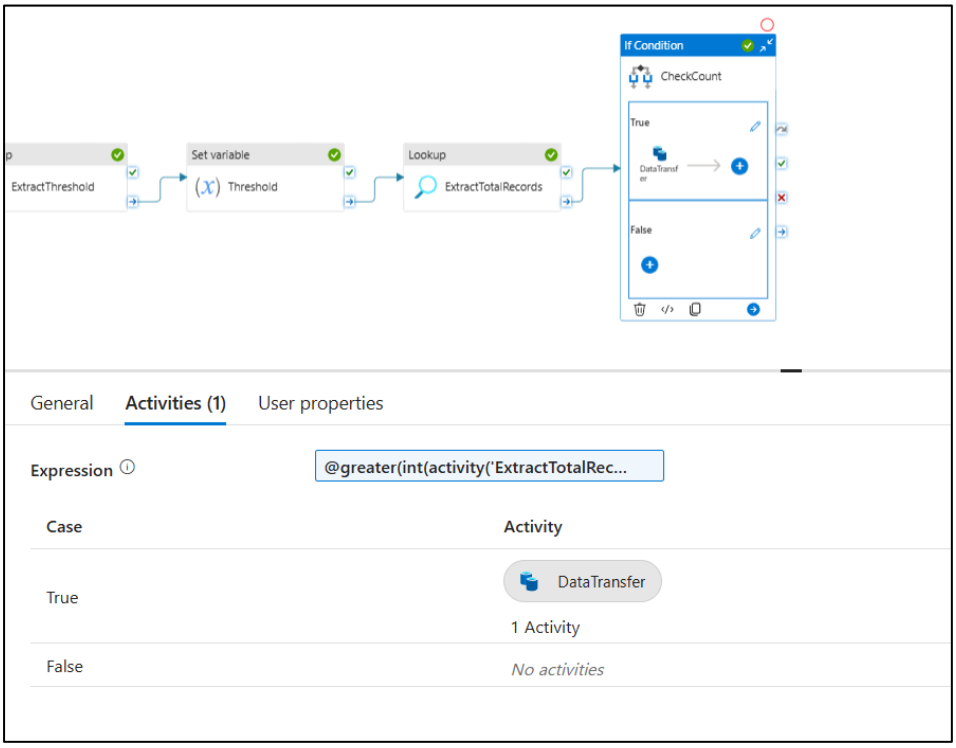
Source Dataset is the customer table(we created earlier, Page 4)

Query: To count the total number of records

SELECT count(*) AS 'totalrecords' FROM dbo.Customer;

6. Insert **If Condition** Activity

   This activity checks, if the total record count is greater than the threshold value or not.

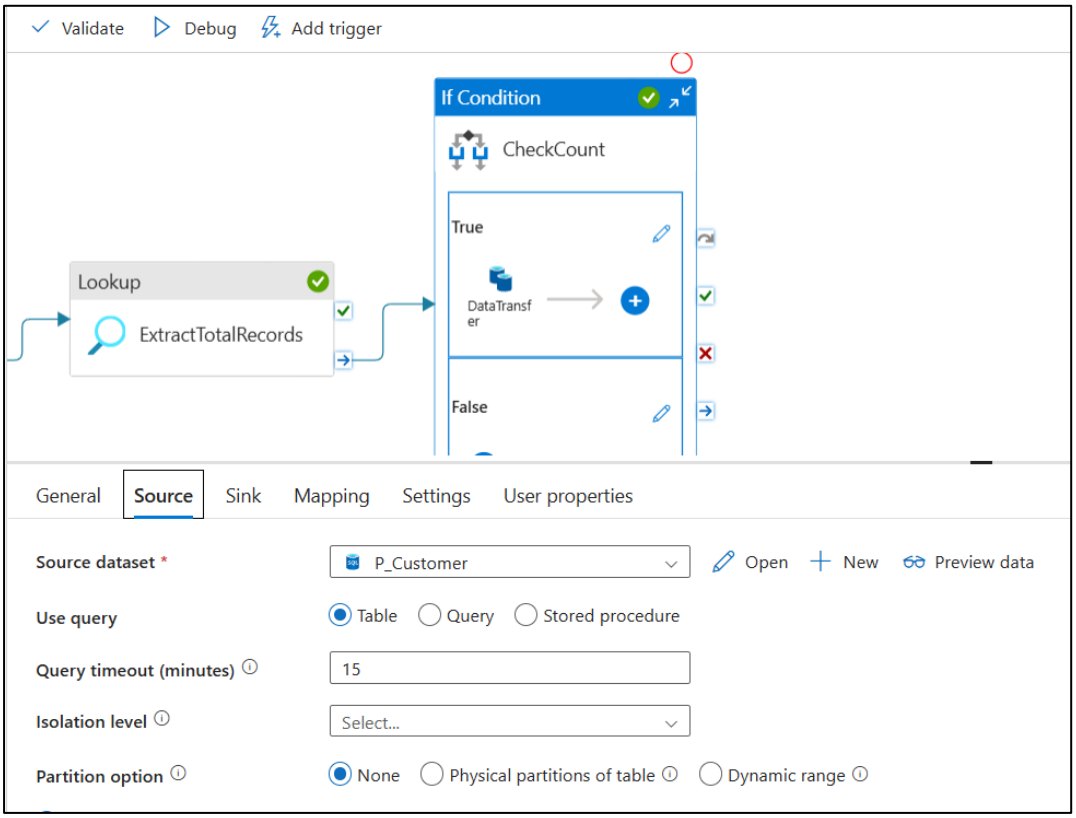


   The expression is:
   @greater(int(activity('ExtractTotalRecords').output.firstRow.totalrecords), int(variables('threshold')))

   It compares the variable threshold and the output from previous activity, i.e, totalRecords from Lookup acitivity.

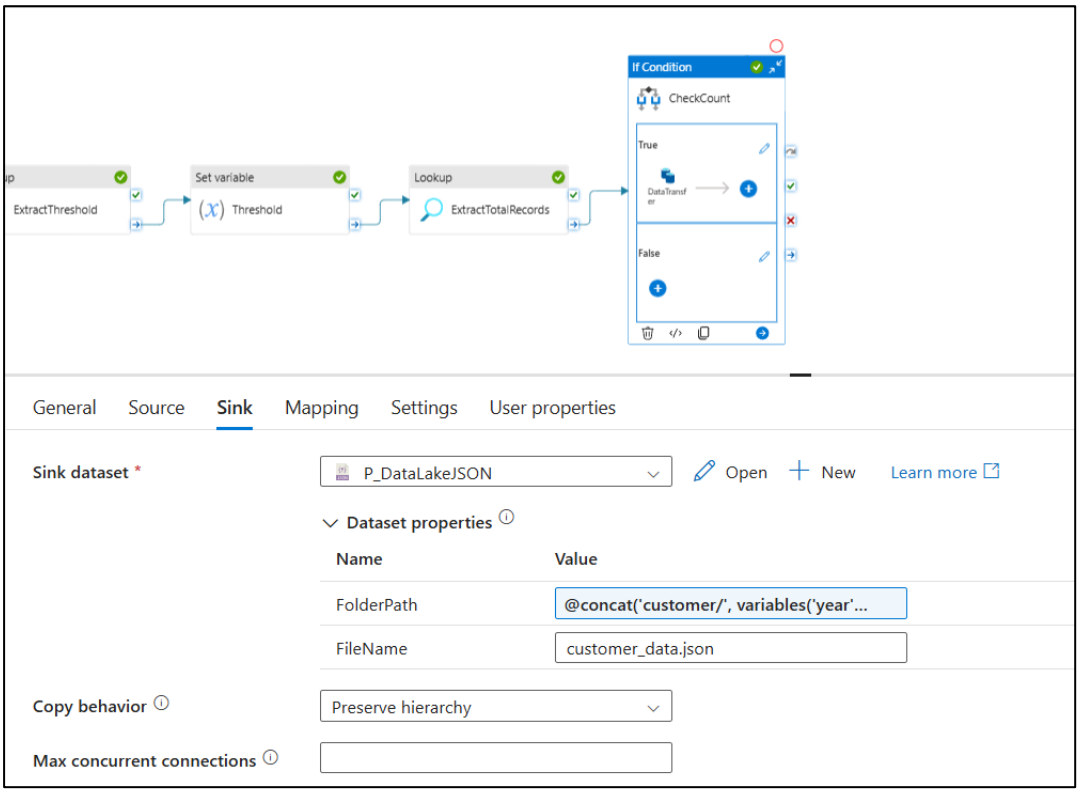
   If the record count is greater than the threshold value, we need to perform data copy operation. Therefore we need to add **Copy Data** activity in the True condition.

7. Insert **Copy Data** activity in True Condition



Where,

Source dataset is the customer table(created on [Page 4](#))



Where,

Sink Dataset is the target file we created earlier(see Page 5)

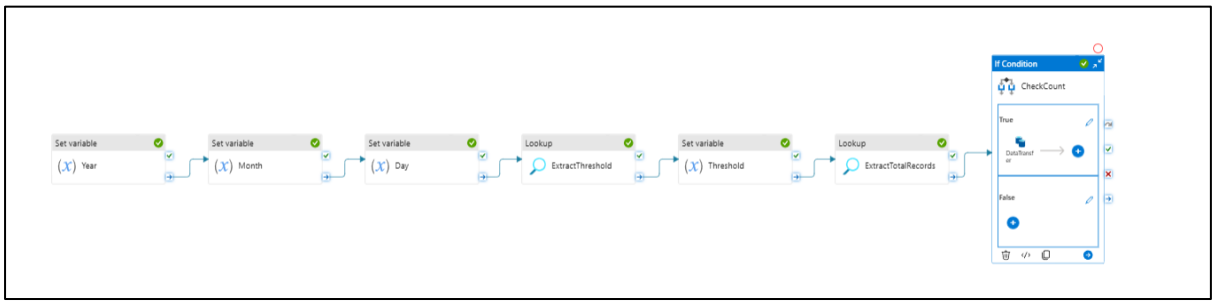
Parameters:

FolderPath: @concat('customer/', variables('year'), '/', variables('month'), '/', variables('day'), '/')

FileNmame: customer_data.json

Preserve the hierarchy and create the file as **Array of objects**

Final Pipeline:



Pipeline Execution

Activity runs

Pipeline run ID 179b8004-dba0-4762-94d4-d98054711c46

All status List Monitor in Azure Metrics

Showing 1 - 8 of 8 items

Activity name	Activity st...	Activit...	Run start	Duration	Integration runtime	User prop...
DataTransfer	Succeeded	Copy data	7/20/2025, 5:44:16 PM	13s	AutoResolveIntegrationRuntime (Central India)	
CheckCount	Succeeded	If Condition	7/20/2025, 5:44:14 PM	18s		
ExtractTotalRecords	Succeeded	Lookup	7/20/2025, 5:43:44 PM	29s	AutoResolveIntegrationRuntime (Central India)	
Threshold	Succeeded	Set variable	7/20/2025, 5:43:43 PM	Less than 1s		
ExtractThreshold	Succeeded	Lookup	7/20/2025, 5:43:28 PM	14s	AutoResolveIntegrationRuntime (Central India)	
Day	Succeeded	Set variable	7/20/2025, 5:43:27 PM	Less than 1s		
Month	Succeeded	Set variable	7/20/2025, 5:43:26 PM	Less than 1s		
Year	Succeeded	Set variable	7/20/2025, 5:43:26 PM	Less than 1s		

Step-Wise Output:

1. Set Variables Activity:
   Variable **year**:

Output

Copy to clipbaord


```
{
    "name": "year",
    "value": "2025"
}
```


Variable **month:**

Output

Copy to clipbaord


```
{
    "name": "month",
    "value": "July"
}
```

Variable **day:**

Output

Copy to clipbord


```
{
    "name": "day",
    "value": "20"
}
```


2. Lookup Acitvity(ExtractThreshold)

Output

Copy to clipbord


```
{
    "firstRow": {
        "threshold": 3
    },
    "effectiveIntegrationRuntime":
"AutoResolveIntegrationRuntime (Central India)",
    "billingReference": {
        "activityType": "PipelineActivity",
        "billableDuration": [
```


3. Set Variable Activity(Threshold)

Output

Copy to clipbord


```
{
    "name": "threshold",
    "value": 3
}
```

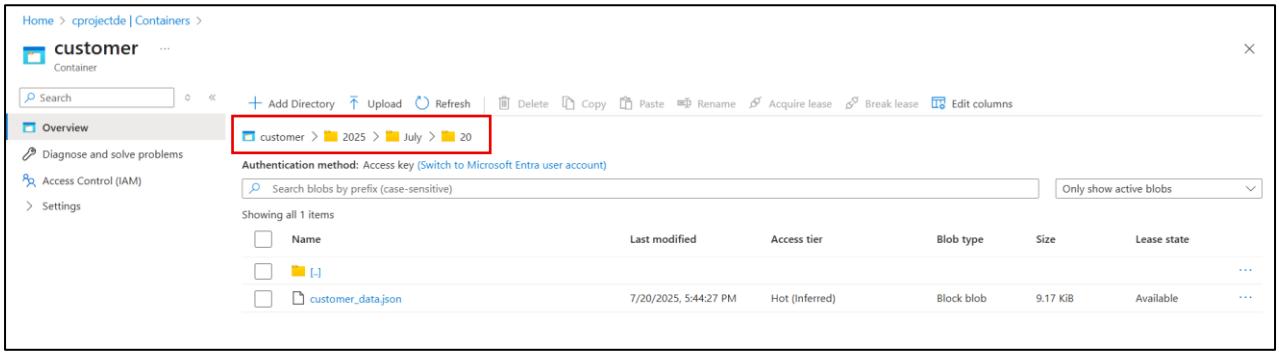
4. Lookup Activity(ExtractTotalRecords)


```
Output
Copy to clipbord

{
    "firstRow": {
        "totalrecords": 71
    },
    "effectiveIntegrationRuntime":
"AutoResolveIntegrationRuntime (Central India)",
    "billingReference": {
        "activityType": "PipelineActivity",
        "billableDuration": [
```


5. Copy Data Activity(DataTransfer)


```
Output
Copy to clipbord  Learn more on output details

{
    "dataRead": 5366,
    "dataWritten": 9387,
    "filesWritten": 1,
    "sourcePeakConnections": 1,
    "sinkPeakConnections": 1,
    "rowsRead": 71,
    "rowsCopied": 71,
    "copyDuration": 10,
```


Target JSON file:



File has been created in **customer/2025/July/20** directory of **cprojectde** storage account.

customer_data.json File:

2025/July/20/customer_data.json

Blob

Save Discard Download Refresh Delete

Overview Versions Snapshots Edit Generate SAS


```
,{"customer_id":150,"first_name":"Ritika","last_name":"Chopra","email":"ritika.chopra@mail.com","created_date":"2025-08-01T00:00:00"
,{"customer_id":175,"first_name":"Sahil","last_name":"Malhotra","email":"sahil.malhotra@mail.com","created_date":"2025-08-02T00:00:0
,{"customer_id":200,"first_name":"Ayesha","last_name":"Khan","email":"ayesha.khan@mail.com","created_date":"2025-08-03T00:00:00"}
,{"customer_id":225,"first_name":"Varun","last_name":"Agarwal","email":"varun.agarwal@mail.com","created_date":"2025-08-04T00:00:00"
,{"customer_id":250,"first_name":"Ishita","last_name":"Bajaj","email":"ishita.bajaj@mail.com","created_date":"2025-08-05T00:00:00"}
,{"customer_id":275,"first_name":"Raj","last_name":"Saxena","email":"raj.saxena@mail.com","created_date":"2025-08-06T00:00:00"}
,{"customer_id":300,"first_name":"Pallavi","last_name":"Nair","email":"pallavi.nair@mail.com","created_date":"2025-08-07T00:00:00"}
,{"customer_id":350,"first_name":"Yuvraj","last_name":"Singhania","email":"yuvraj.singhania@mail.com","created_date":"2025-08-08T00
,{"customer_id":400,"first_name":"Meera","last_name":"Tripathi","email":"meera.tripathi@mail.com","created_date":"2025-08-09T00:00:0
,{"customer_id":450,"first_name":"Nikhil","last_name":"Kulkarni","email":"nikhil.kulkarni@mail.com","created_date":"2025-08-10T00:0
,{"customer_id":1001,"first_name":"Ankit","last_name":"Verma","email":"ankit.verma@mail.in","created_date":"2025-08-17T00:00:00"}
,{"customer_id":1002,"first_name":"Priya","last_name":"Sharma","email":"priya.sharma@mail.in","created_date":"2025-08-18T00:00:00"}
,{"customer_id":1003,"first_name":"Rohit","last_name":"Mehta","email":"rohit.mehta@mail.in","created_date":"2025-08-19T00:00:00"}
,{"customer_id":1004,"first_name":"Sneha","last_name":"Patel","email":"sneha.patel@mail.in","created_date":"2025-08-20T00:00:00"}
,{"customer_id":1005,"first_name":"Arjun","last_name":"Kapoor","email":"arjun.kapoor@mail.in","created_date":"2025-08-21T00:00:00"}
,{"customer_id":1006,"first_name":"Neha","last_name":"Reddy","email":"neha.reddy@mail.in","created_date":"2025-08-22T00:00:00"}
,{"customer_id":1007,"first_name":"Karan","last_name":"Joshi","email":"karan.joshi@mail.in","created_date":"2025-08-23T00:00:00"}
,{"customer_id":1008,"first_name":"Pooja","last_name":"Mishra","email":"pooja.mishra@mail.in","created_date":"2025-08-24T00:00:00"}
,{"customer_id":1009,"first_name":"Manish","last_name":"Gupta","email":"manish.gupta@mail.in","created_date":"2025-08-25T00:00:00"}
,{"customer_id":1010,"first_name":"Divya","last_name":"Singh","email":"divya.singh@mail.in","created_date":"2025-08-26T00:00:00"}
]
```


Json Preview

The pipeline was also executed on 19th July, 2025. After executing pipeline on 20th July 2025, the old data was not overwritten.

customer > 2025 > July

Authentication method: Access key (Switch to Microsoft Entra user account)

Search blobs by prefix (case-sensitive) Only show active blobs

Showing all 4 items

Name	Last modified	Access tier	Blob type	Size	Lease state
[..]					
19					
20					
19	7/19/2025, 11:11:14 PM	Hot (Inferred)	Block blob	0	Available
20	7/20/2025, 5:44:27 PM	Hot (Inferred)	Block blob	0	Available

2025/July/19/customer_data.json

Blob

Save Discard Download Refresh Delete

Overview Versions Snapshots Edit Generate SAS


```
[{"customer_id":1,"first_name":"Rahul","last_name":"Gupta","email":"rahul@mail.com","created_date":"2025-07-11T00:00:00"}
,{"customer_id":2,"first_name":"Prerna","last_name":"Aggarwal","email":"prerna@mail.com","created_date":"2025-07-12T00:00:00"}
,{"customer_id":3,"first_name":"Khushi","last_name":"Goel","email":"khushi@mail.com","created_date":"2025-07-13T00:00:00"}
,{"customer_id":4,"first_name":"Akshat","last_name":"Bansal","email":"akshat@mail.com","created_date":"2025-07-14T00:00:00"}
,{"customer_id":5,"first_name":"Aditya","last_name":"Mittal","email":"aditya@mail.com","created_date":"2025-07-15T00:00:00"}
]
```


There were 5 records when executed on 19th July, 2025.