

## STATEMENT 1

Create a file with threshold value and load this file into your ADLS location. This file contains a threshold value. Create a pipeline in such a manner that, it will first get the threshold value from this file and then check if the record count in the customer table is more than it or not. If yes then copy the customer data from SQL db. to ADLS location in JSON format.

According to the problem statement, we have a **threshold file** in ADLS(Azure Data Lake Storage) location and a **customer table** in Azure SQL. We need to extract threshold value from the ADLS file, check the total number of records present in the customer table, if the record count exceeds the threshold value, we have to copy the table data to ADLS location in JSON format.

### Prerequisites:

1. Threshold file in ADLS location

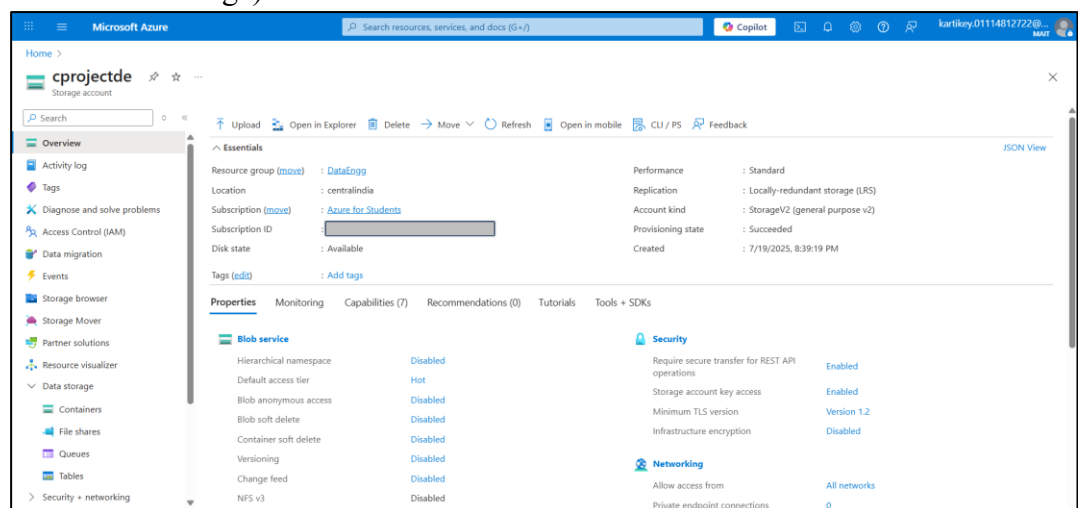
For storing the file in ADLS location, we need to create a **storage account** in Azure. Provide the

Subscription

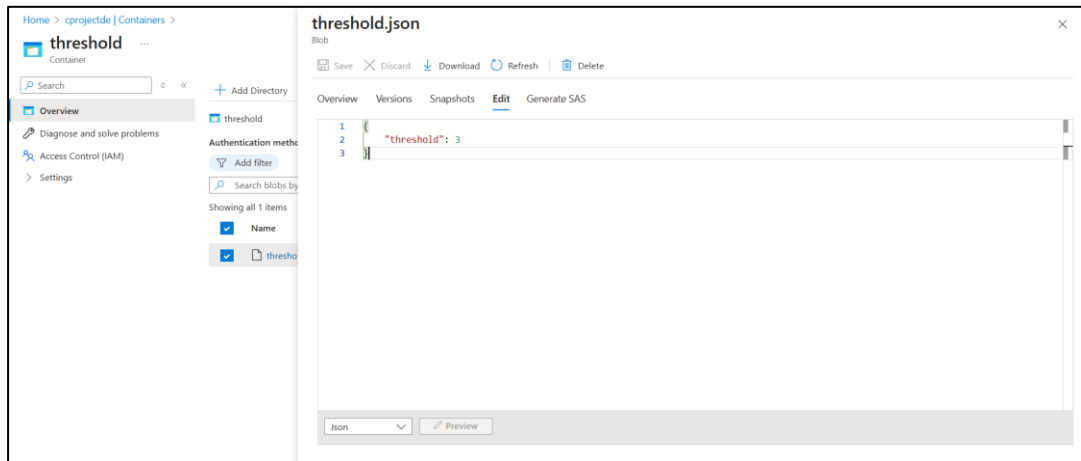
Resource Group

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, 'Perna', 'Aggarwal', 'perna@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 4

## 1. Threshold File Dataset

Create **Azure Data Lake Storage Gen2** dataset.

The format of the dataset should be **JSON**

JSON  
P\_DataLakeJSON

Connection Schema Parameters

Linked service \* P\_AzureDataLakeStorage Test connection Edit + New Learn more

File path @dataset().FolderPath / Directory / @dataset().FileName Browse Preview data

Compression type No compression

Encoding Default(UTF-8)

Create two parameters:

JSON  
P\_DataLakeJSON

Connection Schema Parameters

+ New Delete

<input type="checkbox"/>	Name	Type	Default value	
<input type="checkbox"/>	FolderPath	String	Value	
<input type="checkbox"/>	FileName	String	Value	

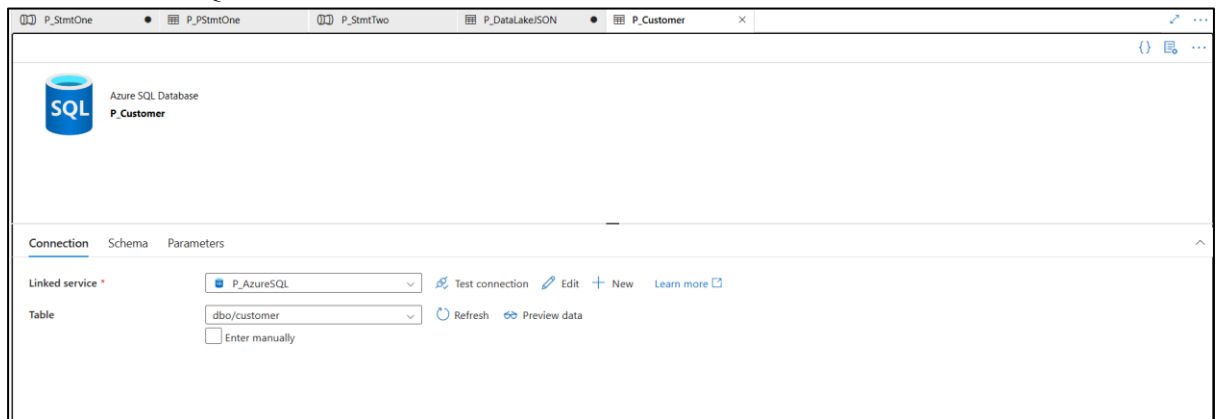
Populate the **File Path**:

**File System:** @dataset().FolderPath

**File Name:** @dataset().FileName

## 2. SQL Table Dataset

Create Azure SQL dataset



Provide:

Dataset Name

Linked Service

Table Name

**Set properties**

**Name**  
DatasetName

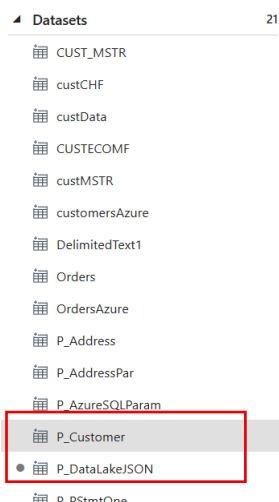
**Linked service \***  
P\_AzureSQL

**Table name**  
dbo.customer

☐ Enter manually

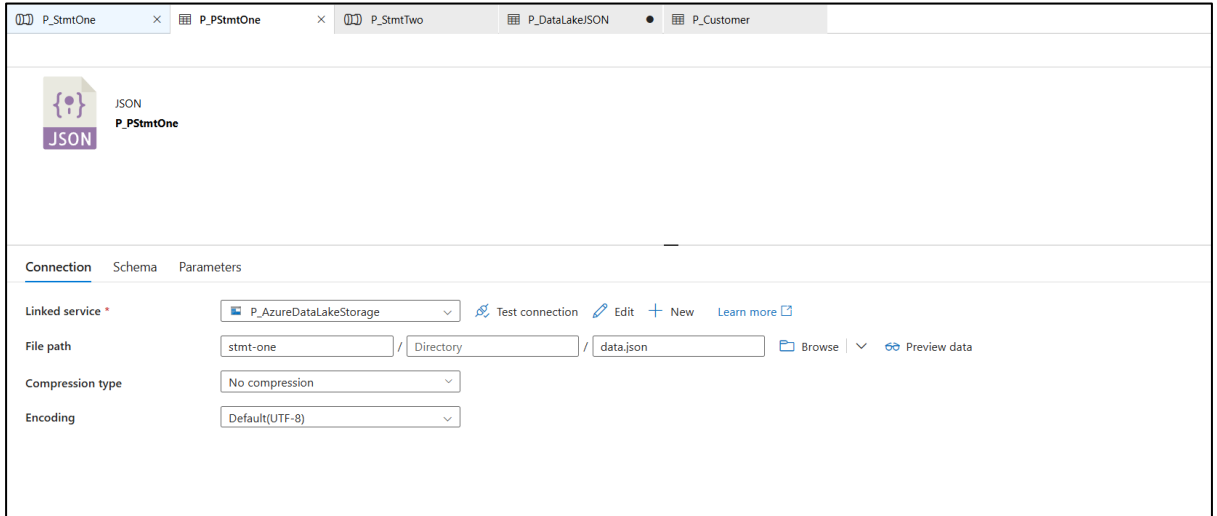
**Import schema**  
☒ From connection/store ☐ None

Final



### 3. Target File

Create target JSON file dataset, where the data will be stored.



The screenshot shows the configuration interface for a JSON file dataset in Azure Data Lake Storage. The top bar displays several tabs: 'P\_StmtOne', 'P\_PStmtOne', 'P\_StmtTwo', 'P\_DataLakeJSON', and 'P\_Customer'. The main area features a JSON icon and the text 'JSON P\_PStmtOne'. Below this, there are three tabs: 'Connection', 'Schema', and 'Parameters'. The 'Connection' tab is active, showing the following configuration:

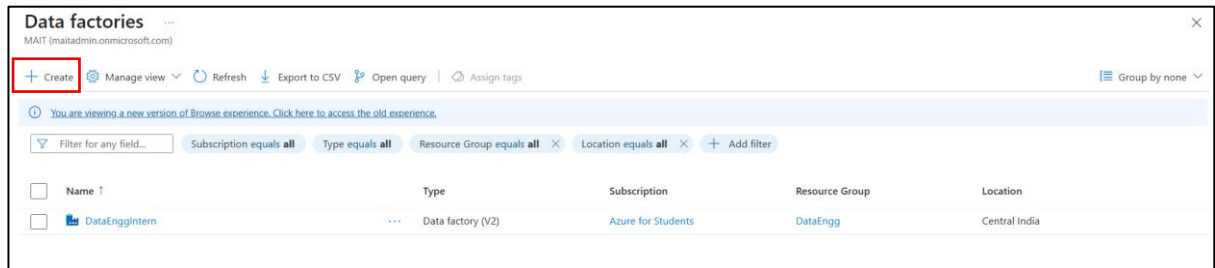
- Linked service:** A dropdown menu set to 'P\_AzureDataLakeStorage'. To the right are links for 'Test connection', 'Edit', '+ New', and 'Learn more'.
- File path:** A text input field containing 'stmt-one', followed by a 'Directory' label and a slash, and another text input field containing 'data.json'. To the right are 'Browse' and 'Preview data' links.
- Compression type:** A dropdown menu set to 'No compression'.
- Encoding:** A dropdown menu set to 'Default(UTF-8)'.

As shown in the above image, the data will be stored in the directory **stmt-one** in file named **data.json**. This directory is present in the cprojectde storage account we created earlier.

## 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.



Home > Data factories >

## Create Data Factory

**Basics** | Git configuration | Networking | Advanced | Tags | Review + create

One-click to create data factory with sample pipeline and datasets. [Try it](#)

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

Resource group \* ⓘ

[Create new](#)

### Instance details

Name \* ⓘ

Region \* ⓘ

Version \* ⓘ

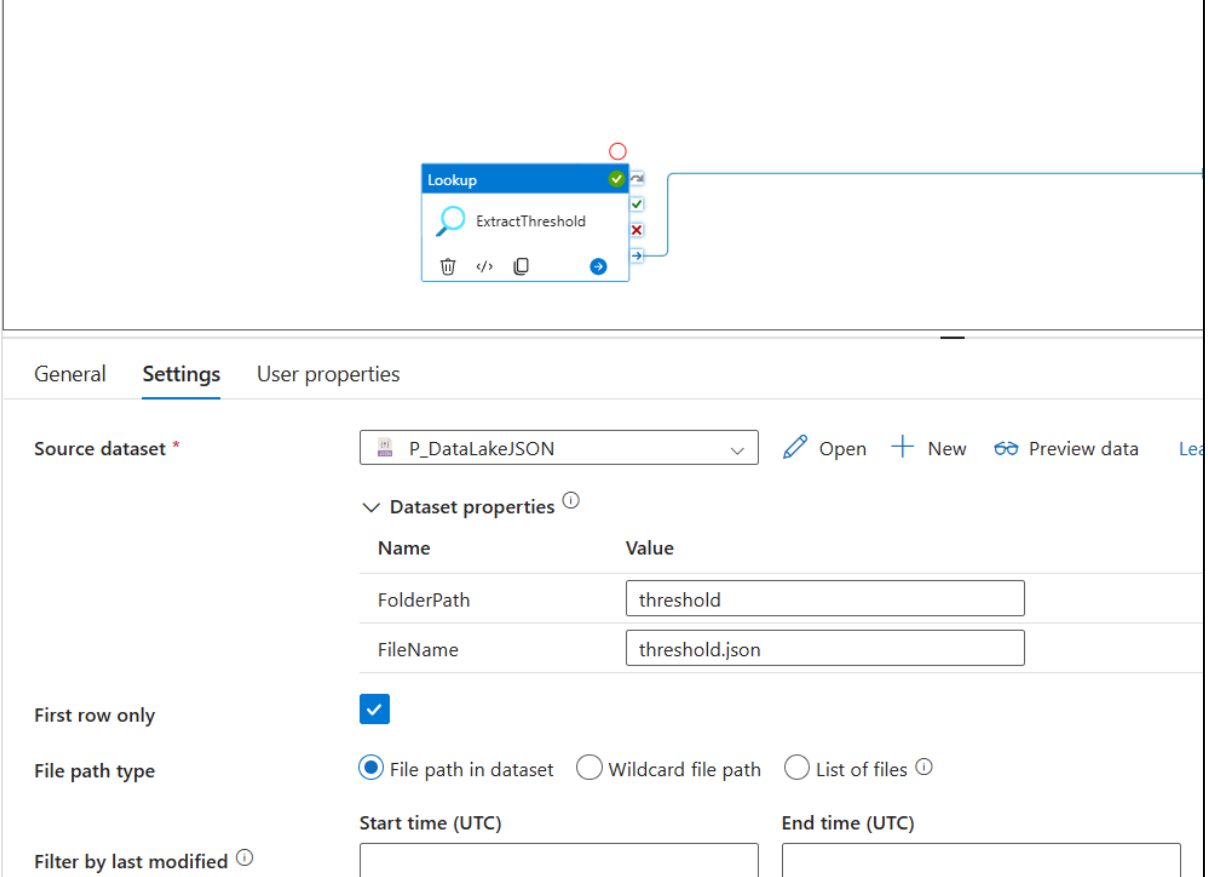
Azure for Students

East US

V2

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

1. Create new pipeline, name it accordingly.
2. Insert **Lookup** Activity



The screenshot shows the configuration of a **Lookup** activity named **ExtractThreshold**. The activity is connected to a dataset named **P\_DataLakeJSON**. The configuration is as follows:

Name	Value
FolderPath	threshold
FileName	threshold.json

Additional settings:

- First row only:** ☒
- File path type:** ☒ File path in dataset ☐ Wildcard file path ☐ List of files
- Filter by last modified:**  (Start time (UTC))  (End time (UTC))

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 directory in which json file is present(can be left blank if file is not in directory)

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

### 3. Insert **Set Variable** activity

The screenshot shows a pipeline editor with two activities: 'Lookup' and 'Set variable'. The 'Lookup' activity is named 'ExtractThreshold' and has a green checkmark. It is connected to the 'Set variable' activity, which is named 'Threshold' and also has a green checkmark. The 'Set variable' activity is configured with the variable name 'threshold' and the value '@activity('ExtractThreshold').output.f...'. Below the activities, the 'Settings' tab is selected, showing the 'Variable type' as 'Pipeline variable' and the 'Name' as 'threshold'. The 'Value' is set to '@activity('ExtractThreshold').output.f...'.

**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**)

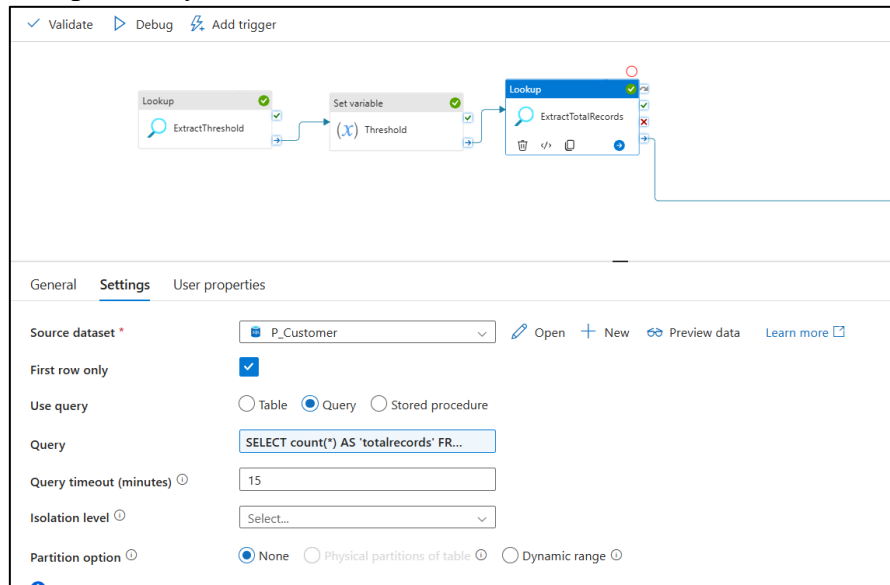
For this, initialise variable 'threshold' in the pipeline of type integer.

The screenshot shows the 'Variables' tab in the pipeline editor. It displays a table with columns for 'Name', 'Type', and 'Default value'. A variable named 'threshold' is listed with the type 'Integer' and a default value of 'Value'.

Name	Type	Default value
threshold	Integer	Value



#### 4. Insert **Lookup** Activity



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

Where,

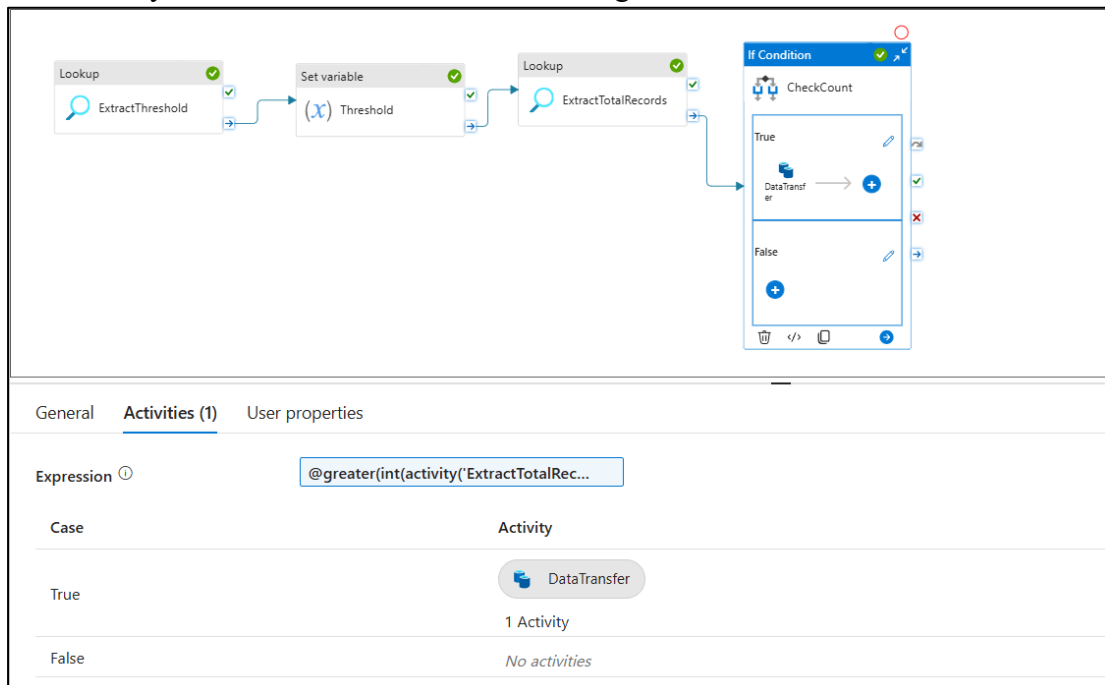
Source Dataset is the customer table(we created earlier, [Page 4](#))

Query: To count the total number of records

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

#### 5. 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 activity.

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.

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

The screenshot displays the Azure Data Factory (ADF) designer interface. At the top, there are tabs for 'Validate', 'Debug', and 'Add trigger'. The main canvas shows a workflow starting with a 'Lookup' activity named 'ExtractTotalRecords'. An arrow connects this activity to an 'If Condition' activity. The 'If Condition' activity has a condition named 'CheckCount'. Inside the 'If Condition' activity, there are two paths: 'True' and 'False'. The 'True' path contains a 'DataTransfer' activity, which is a 'Copy Data' activity. The 'False' path is currently empty. Below the canvas, the 'Source' tab is selected, showing the configuration for the 'DataTransfer' activity. The 'Source dataset' is set to 'P\_Customer'. The 'Use query' section has 'Table' selected. The 'Query timeout (minutes)' is set to 15. The 'Isolation level' is set to 'Select...'. The 'Partition option' has 'None' selected.

✓ Validate ▶ Debug ⚡ Add trigger

If Condition ✓ ↶ ↷

CheckCount

True

DataTransfer → +

False

General **Source** Sink Mapping Settings User properties

Source dataset \* P\_Customer Open + New Preview data

Use query ☒ Table ☐ Query ☐ Stored procedure

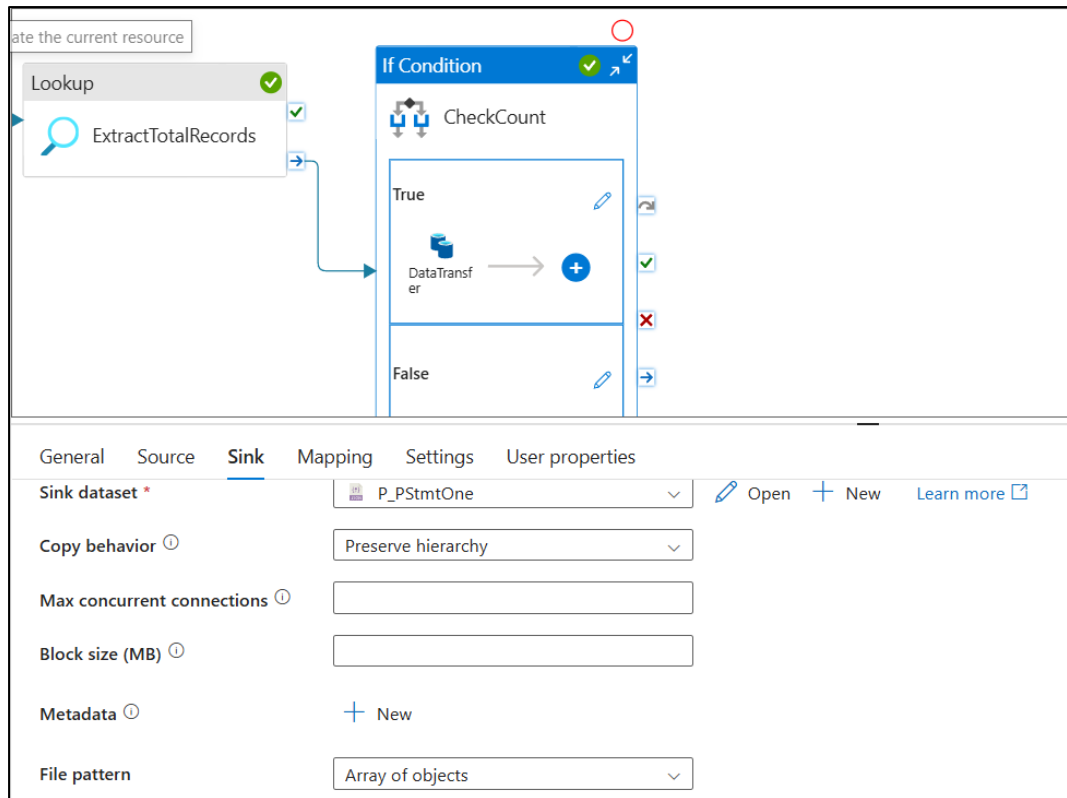
Query timeout (minutes) ⓘ 15

Isolation level ⓘ Select...

Partition option ⓘ ☒ None ☐ Physical partitions of table ⓘ ☐ Dynamic range ⓘ

Where,

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

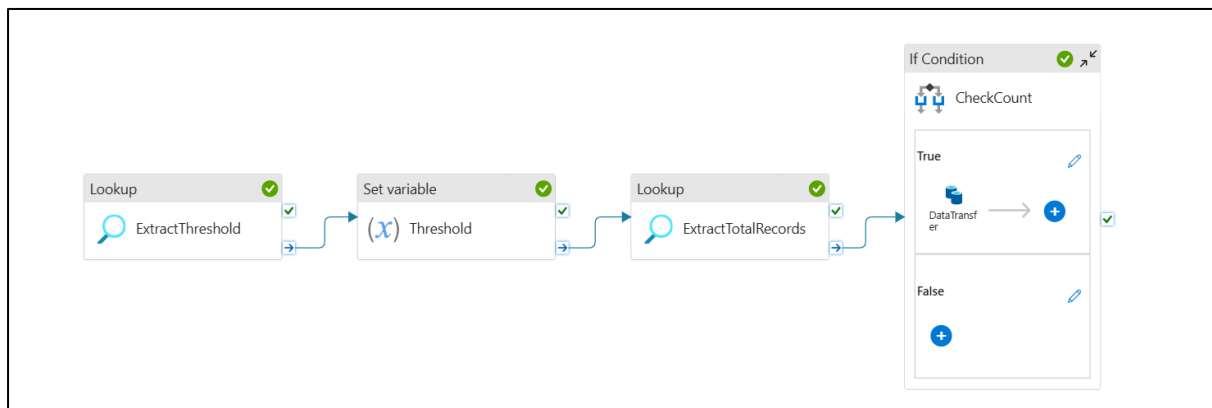


Where,

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

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

## Final Pipeline:



## Pipeline Execution

Activity runs

Pipeline run ID f2e5afb4-a9be-4fd9-9cc4-c776ca3b84ad

All status 

Monitor in Azure Metrics [↗](#) [↓](#)

Showing 1 - 5 of 5 items

Activity name	Activity st...	Activit...	Run start	Duration	Integration runtime	User prop...
DataTransfer	✔ Succeeded	Copy data	7/20/2025, 2:12:41 PM	13s	AutoResolveIntegrationRuntime (Central India)	
CheckCount	✔ Succeeded	If Condition	7/20/2025, 2:12:41 PM	15s		
ExtractTotalRecords	✔ Succeeded	Lookup	7/20/2025, 2:12:26 PM	14s	AutoResolveIntegrationRuntime (Central India)	
Threshold	✔ Succeeded	Set variable	7/20/2025, 2:12:26 PM	Less than 1s		
ExtractThreshold	✔ Succeeded	Lookup	7/20/2025, 2:12:11 PM	14s	AutoResolveIntegrationRuntime (Central India)	

## Step-Wise Output:

### 1. Lookup Acitivity(ExtractThreshold)

Output

Copy to clipboard

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

### 2. Set Variable Activity(Threshold)

Output

Copy to clipboard

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

3. Lookup Activity(ExtractTotalRecords)

Output

Copy to clipboard

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

4. Copy Data Activity(DataTransfer)

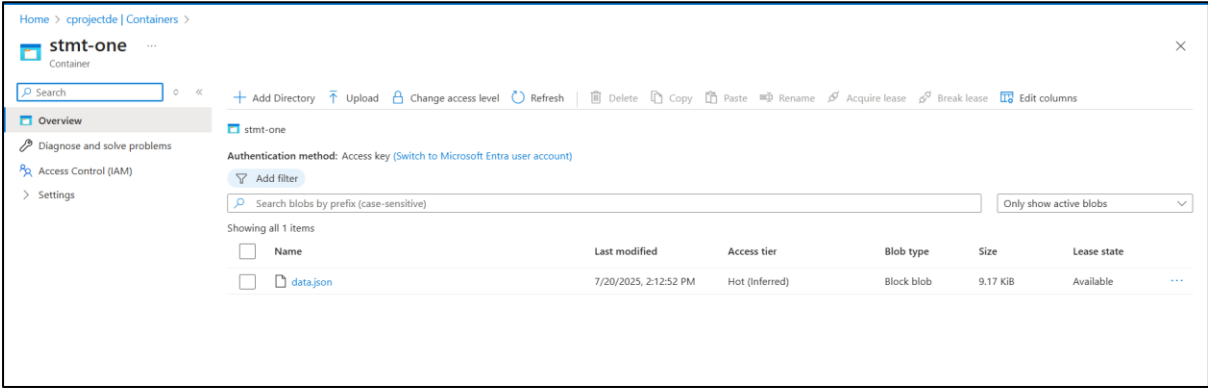
Output

Copy to clipboard

Learn more on output details

```
{
  "dataRead": 5366,
  "dataWritten": 9387,
  "filesWritten": 1,
  "sourcePeakConnections": 1,
  "sinkPeakConnections": 1,
  "rowsRead": 71,
  "rowsCopied": 71,
  "copyDuration": 9,
```

Target JSON file:



File has been created in **stmt-one** directory of **cprojectde** storage account.

Data.json File:

data.json

Blob

Save

Discard

Download

Refresh

Delete

Overview

Versions

Snapshots

Edit

Generate SAS

52

{ "customer\_id":150,"first\_name":"Ritika","last\_name":"Chopra","email":"ritika.chopra@mail.com","created\_date":"2025-08-01T00:00:00"}

53

{ "customer\_id":175,"first\_name":"Sahil","last\_name":"Malhotra","email":"sahil.malhotra@mail.com","created\_date":"2025-08-02T00:00:00"}

54

{ "customer\_id":200,"first\_name":"Ayesha","last\_name":"Khan","email":"ayesha.khan@mail.com","created\_date":"2025-08-03T00:00:00"}

55

{ "customer\_id":225,"first\_name":"Varun","last\_name":"Agarwal","email":"varun.agarwal@mail.com","created\_date":"2025-08-04T00:00:00"}

56

{ "customer\_id":250,"first\_name":"Ishita","last\_name":"Bajaj","email":"ishita.bajaj@mail.com","created\_date":"2025-08-05T00:00:00"}

57

{ "customer\_id":275,"first\_name":"Raj","last\_name":"Saxena","email":"raj.saxena@mail.com","created\_date":"2025-08-06T00:00:00"}

58

{ "customer\_id":300,"first\_name":"Pallavi","last\_name":"Nair","email":"pallavi.nair@mail.com","created\_date":"2025-08-07T00:00:00"}

59

{ "customer\_id":350,"first\_name":"Yuvraj","last\_name":"Singhania","email":"yuvraj.singhania@mail.com","created\_date":"2025-08-08T00:00:00"}

60

{ "customer\_id":400,"first\_name":"Meera","last\_name":"Tripathi","email":"meera.tripathi@mail.com","created\_date":"2025-08-09T00:00:00"}

61

{ "customer\_id":450,"first\_name":"Nikhil","last\_name":"Kulkarni","email":"nikhil.kulkarni@mail.com","created\_date":"2025-08-10T00:00:00"}

62

{ "customer\_id":1001,"first\_name":"Ankit","last\_name":"Verma","email":"ankit.verma@mail.in","created\_date":"2025-08-17T00:00:00"}

63

{ "customer\_id":1002,"first\_name":"Priya","last\_name":"Sharma","email":"priya.sharma@mail.in","created\_date":"2025-08-18T00:00:00"}

64

{ "customer\_id":1003,"first\_name":"Rohit","last\_name":"Mehta","email":"rohit.mehta@mail.in","created\_date":"2025-08-19T00:00:00"}

65

{ "customer\_id":1004,"first\_name":"Sneha","last\_name":"Patel","email":"sneha.patel@mail.in","created\_date":"2025-08-20T00:00:00"}

66

{ "customer\_id":1005,"first\_name":"Arjun","last\_name":"Kapoor","email":"arjun.kapoor@mail.in","created\_date":"2025-08-21T00:00:00"}

67

{ "customer\_id":1006,"first\_name":"Neha","last\_name":"Reddy","email":"neha.reddy@mail.in","created\_date":"2025-08-22T00:00:00"}

68

{ "customer\_id":1007,"first\_name":"Karan","last\_name":"Joshi","email":"karan.joshi@mail.in","created\_date":"2025-08-23T00:00:00"}

69

{ "customer\_id":1008,"first\_name":"Pooja","last\_name":"Mishra","email":"pooja.mishra@mail.in","created\_date":"2025-08-24T00:00:00"}

70

{ "customer\_id":1009,"first\_name":"Manish","last\_name":"Gupta","email":"manish.gupta@mail.in","created\_date":"2025-08-25T00:00:00"}

71

{ "customer\_id":1010,"first\_name":"Divya","last\_name":"Singh","email":"divya.singh@mail.in","created\_date":"2025-08-26T00:00:00"}

72

Json

Preview