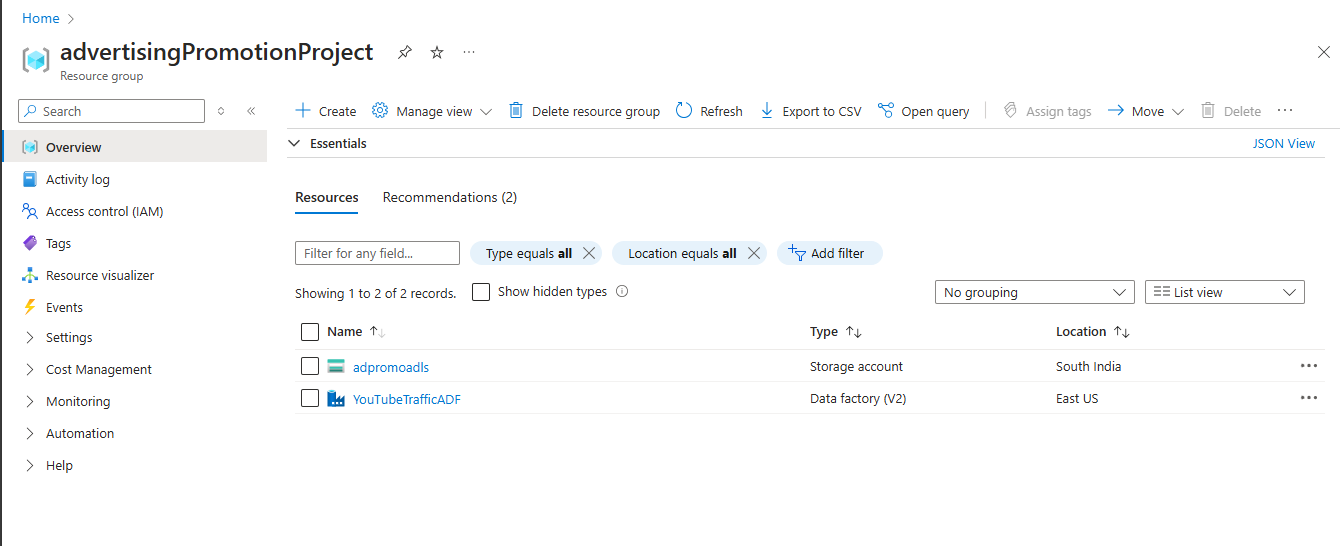
Advertising Promotion Project

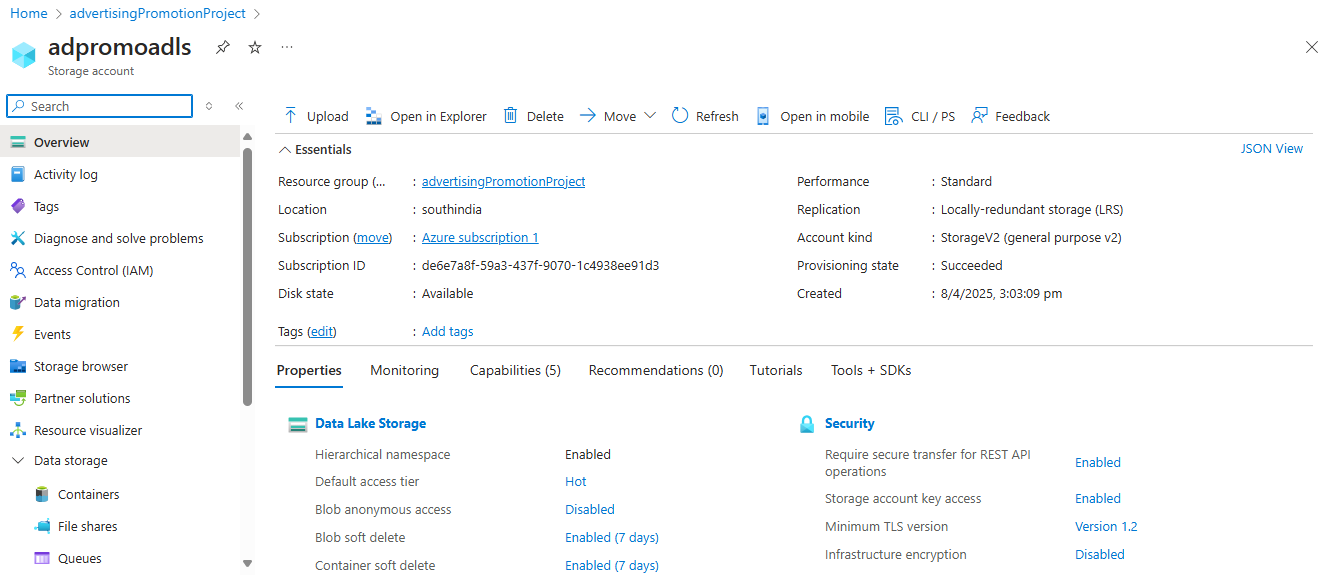
Step1: Create a Resource groups for this project.

A resource group is a logical container that holds related resources for an Azure solution, allowing you to manage them as a single entity, based on lifecycle and security.



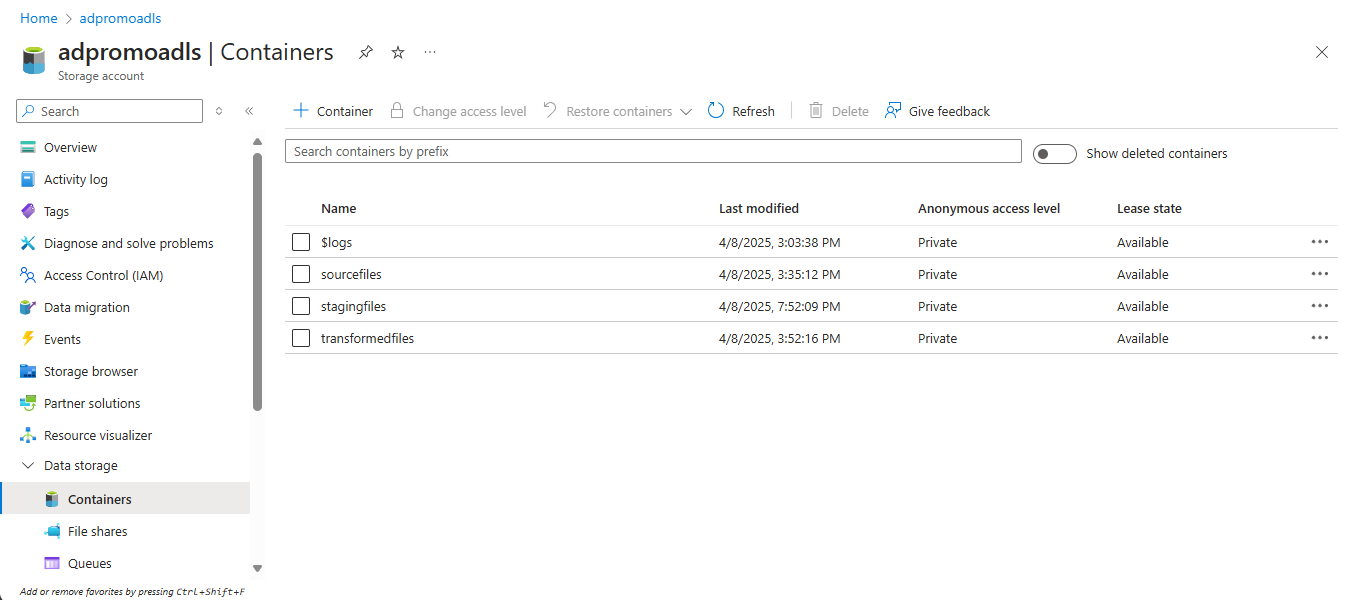
Step2: Create a Storage Account.

Where the Datasets files and transformed files as placed. Since it is mentioned that “Multiple files can be stored at the dynamic location of Azure Data Lake Store and the same needs to transform and copied to any data store.”



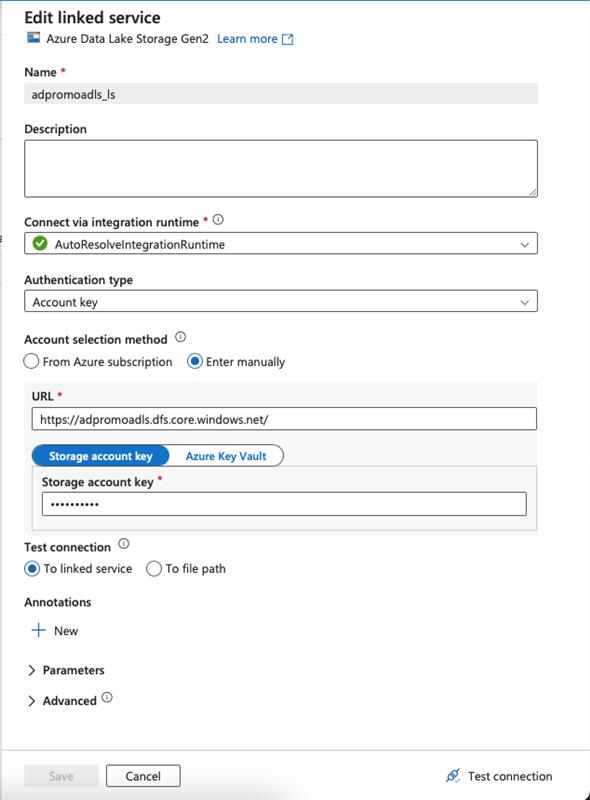
In ADLS, created 3 folders:

* sourcefiles: Source datasets are placed.
* stagingfiles: Before transforming the data, files are moved to stagingfiles.
* transformedfiles: Final data which is transformed and ready to use for visualization.



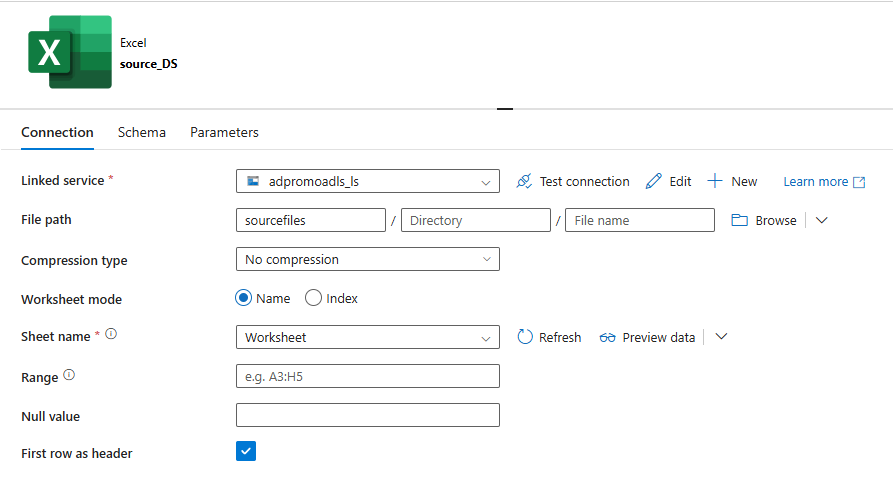
Step3: Create an Azure Data Factory Resource.

1. First, create a linked service for Storage Account.

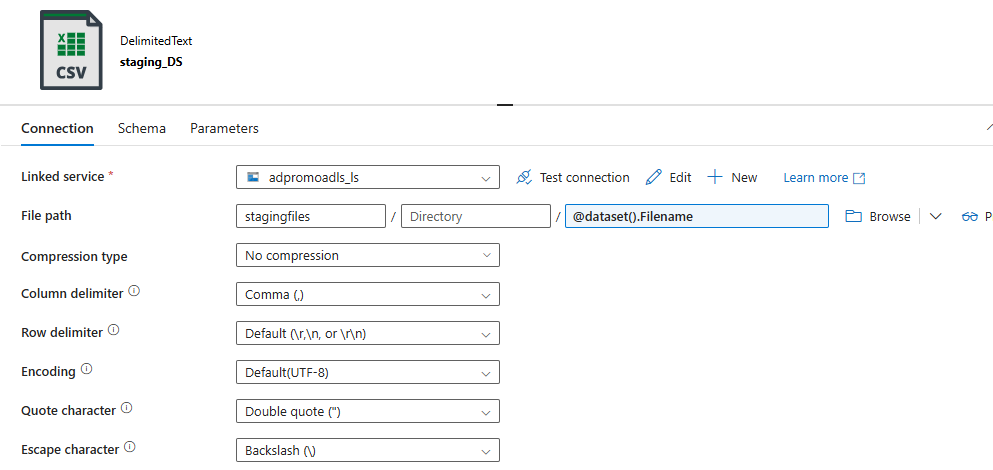


Linked Service defines the connection to external resources, such as data storage or computing services. They are similar to connection strings, allowing ADF to know where to retrieve or send data.

1. Create a Datasets for source data i.e., source\_DS: I have mentioned only folder name value here.

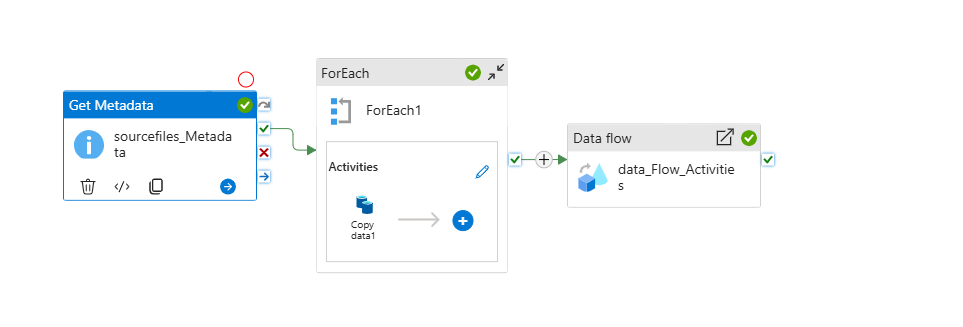


1. Create a Datasets for Stage data i.e., staging\_DS: here, mentioned folder name and dynamic parameter for filename.



1. Create a pipeline:

In pipeline, there are 4 activities to be performed.



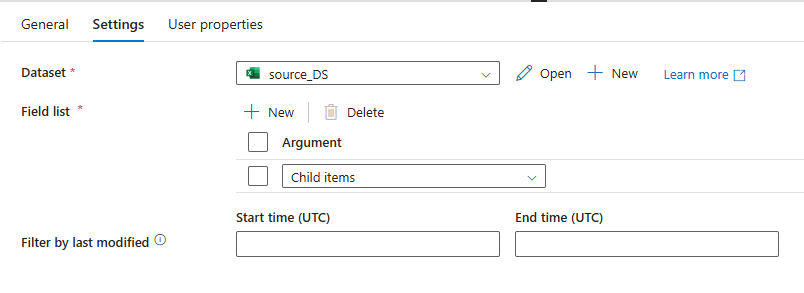
1. Get Metadata

In Azure Data Factory (ADF), the "Get Metadata" activity is a control flow activity that allows you to retrieve metadata about datasets or files from various data stores, without actually moving the data, enabling dynamic pipeline execution and data validation.

* In General, provide the Name for Get Metadata.
* In Setting, provide Dataset value i.e., source\_DS.

Field list as child items.

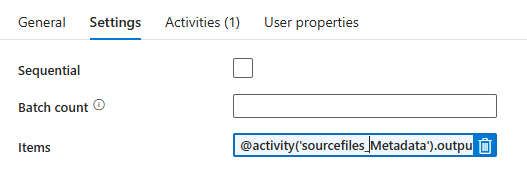
The "childItems" metadata option, when used on a folder, returns a list of all subfolders and files contained within that folder, along with their names and types.

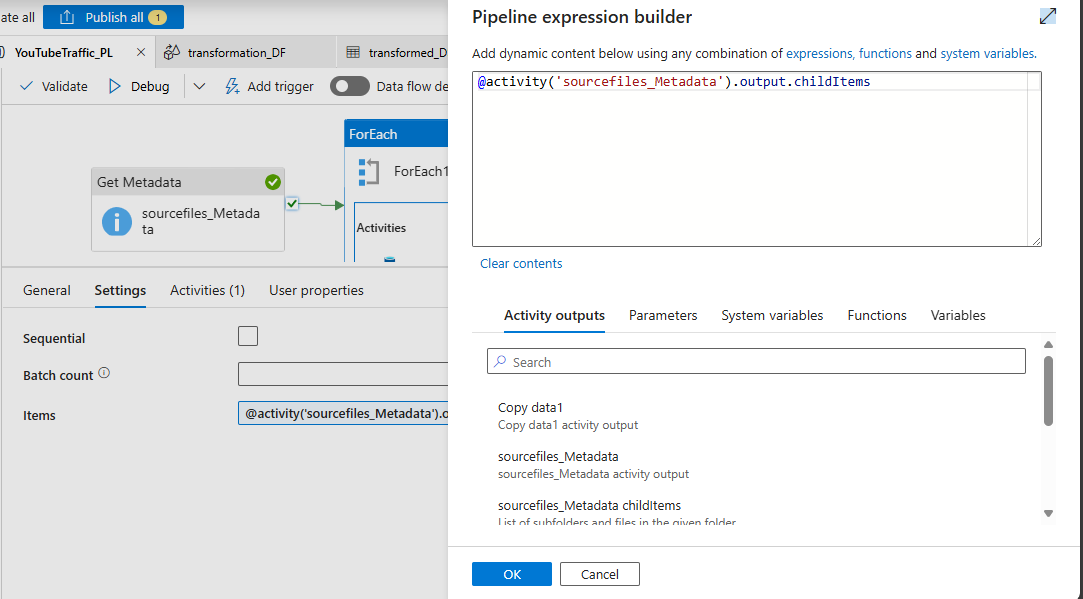


1. ForEach

ForEach acts as a for loop on the value outputted from Get MetaData. Since it returns the value in array.

* In General, provide the Name for ForEach.
* In Setting, we need to populate Items field. We will be getting filenames through Get Metadata activity, so we have to add a dynamic content.

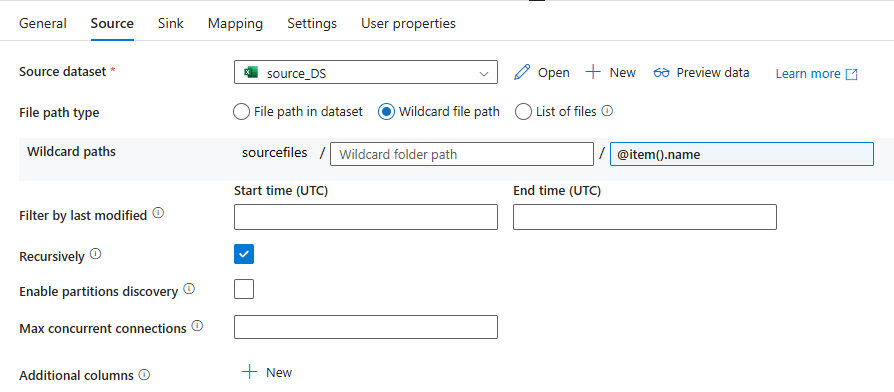


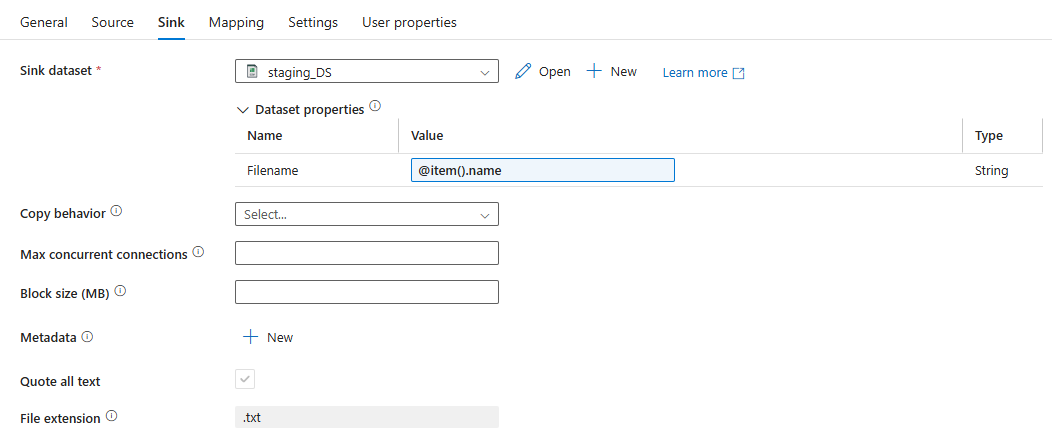


* Inside the ForEach, we need to add another activity i.e., copy data activity. Where we will be copying the files one after the another into a staging folder.

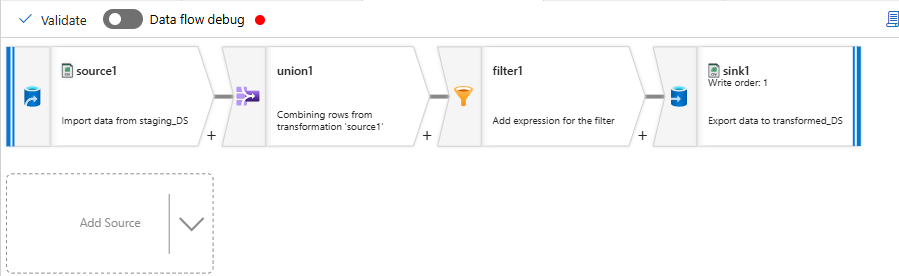
In copy data activity:

In General, provide the Name for copy data.

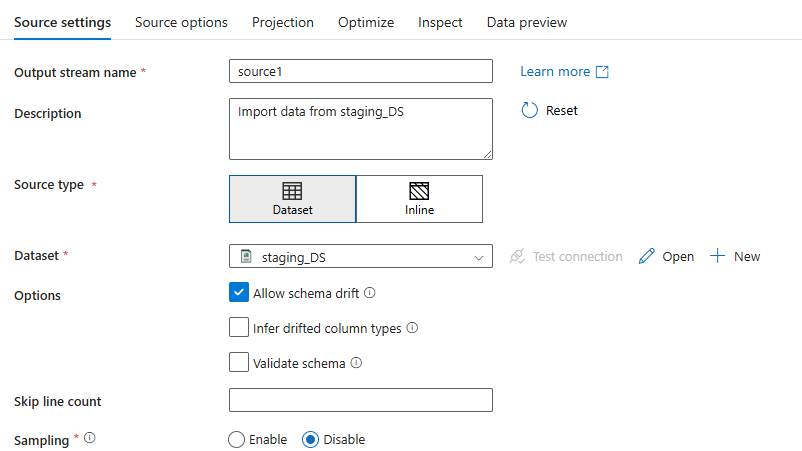




1. Create a Data Flows:

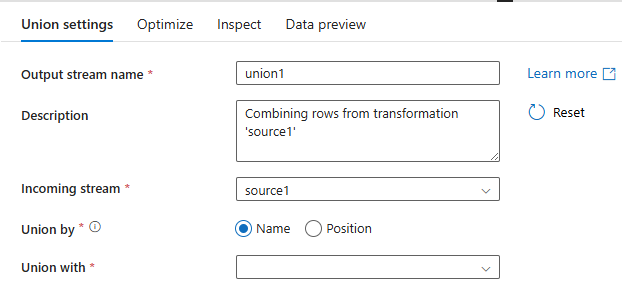


source1:



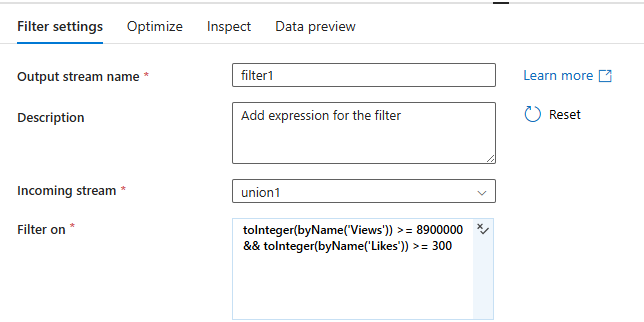
Dataset – considering data from staging\_DS.

union1:



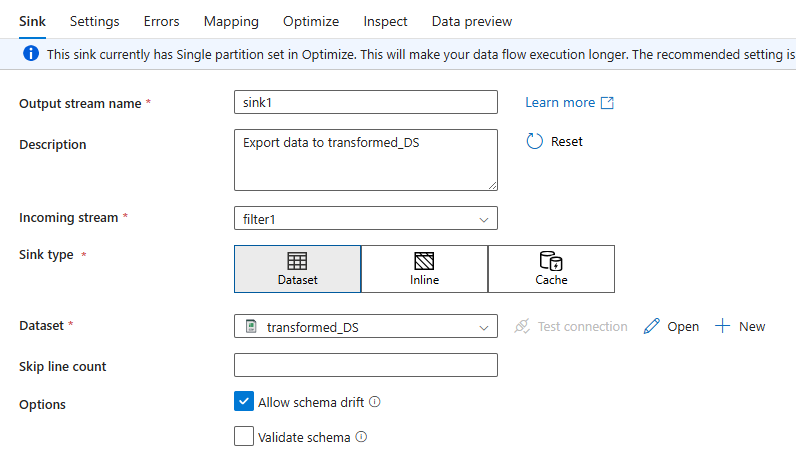
union1 will merge the data if there is more than 1 files and pass it to filter1.

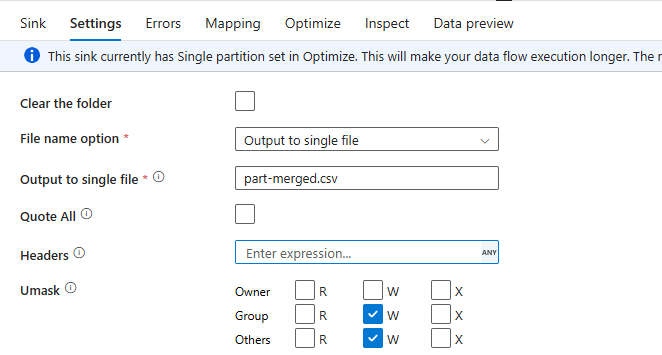
filter1:

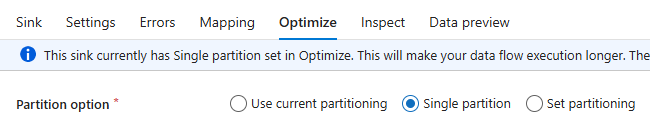


The traffic should be analyzed based on the views. But the minimum views must be 89,00,000 and minimum likes must be 300. To achieve this we have to filter the data according, in filter on pass the following expression “toInteger(byName('Views')) >= 8900000 && toInteger(byName('Likes')) >= 300”.

sink1:







In sink,

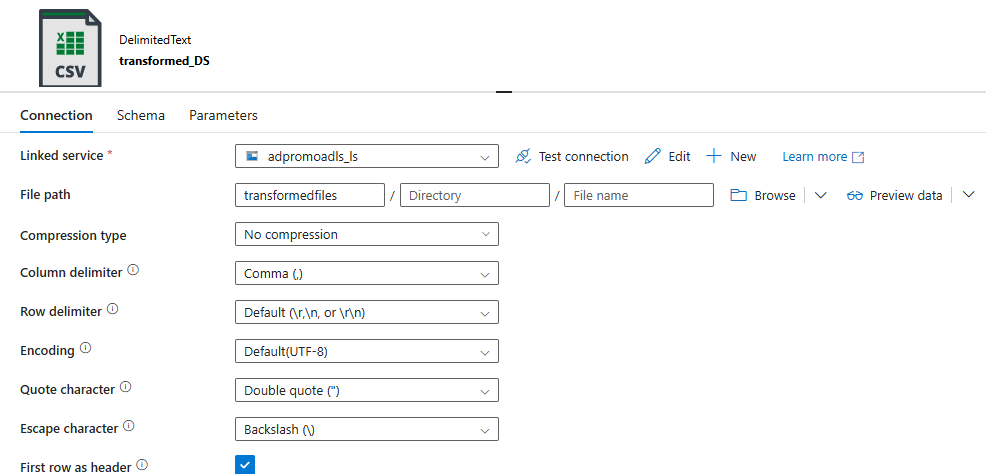
Incoming stream 🡪filter1

Dataset 🡪transformed\_DS

File name option 🡪 output to single file

Output to single file 🡪 part-merged.csv

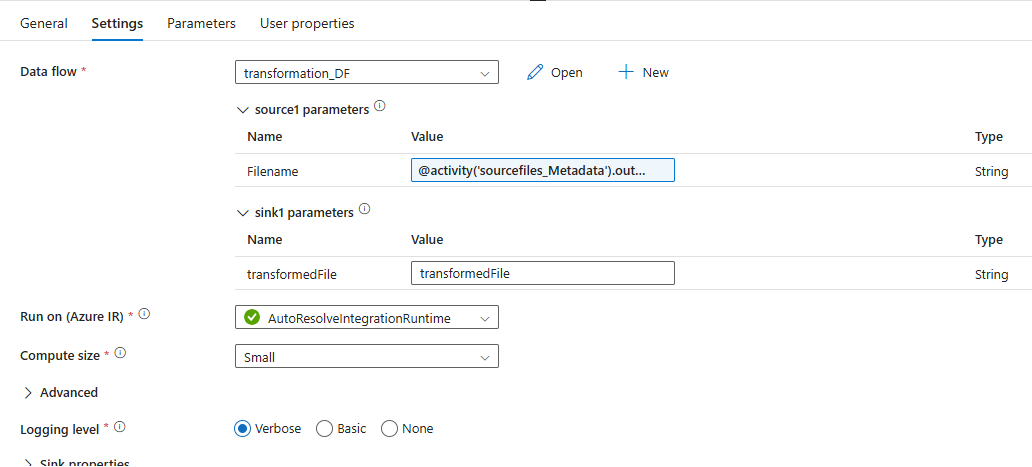
Partition option 🡪 Single partition

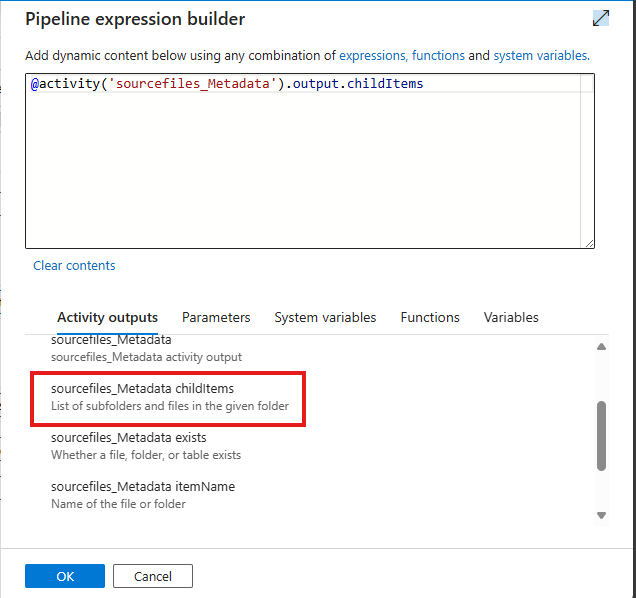


In transformed\_DS:

* Linked service 🡪 adpromoadls\_ls (ADLS Gen 2)
* File path 🡪 transformedfiles (folder)
* First row as header 🡪 Enabled

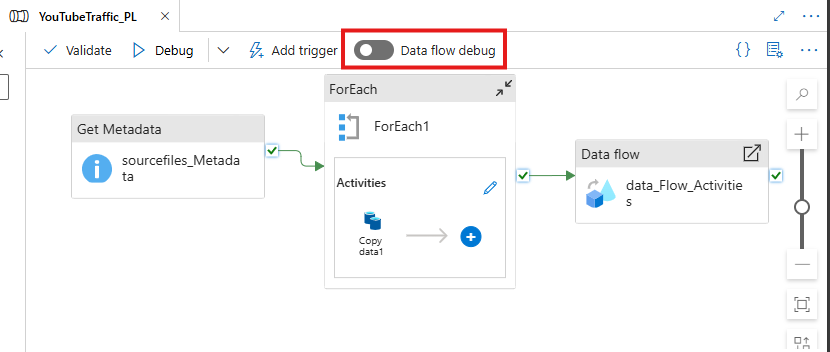
1. Data flow activity:

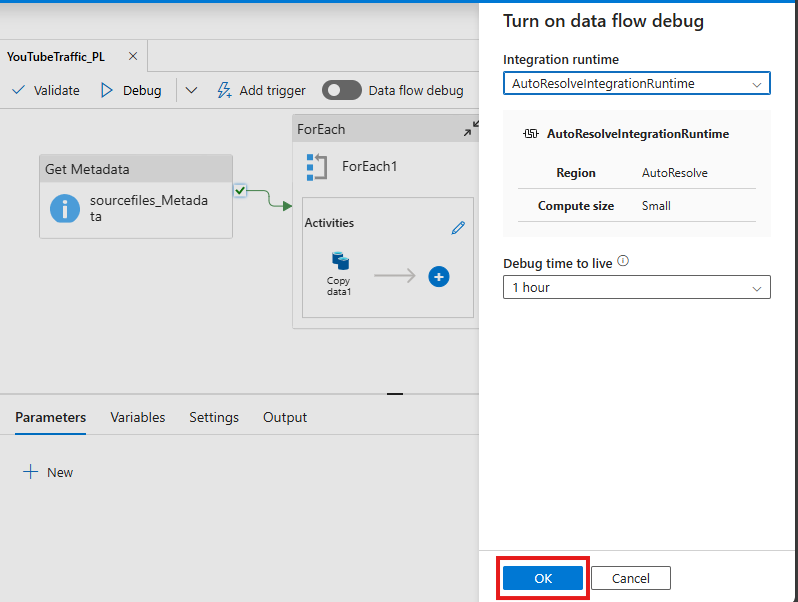




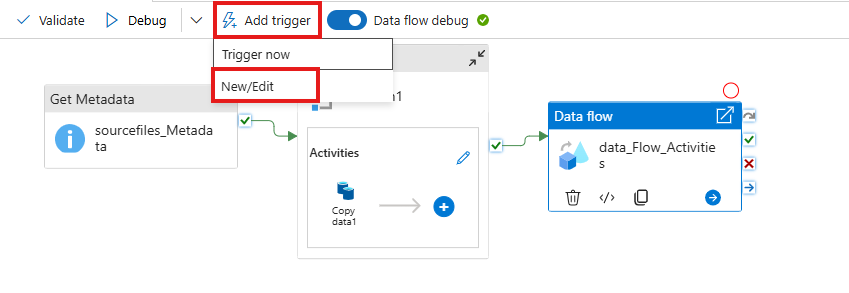
Step4 - Enable Data flow debug:

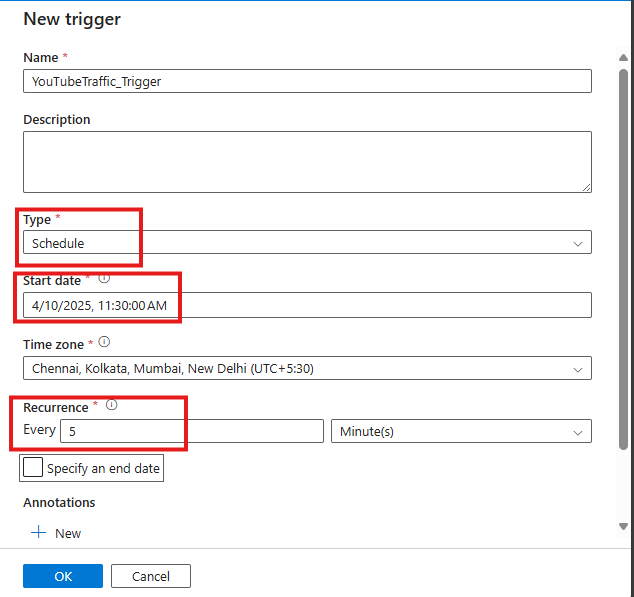
data flow debug mode allows to interactively watch the data shape and transformation logic while building and debugging the data flows, using a live Spark cluster.



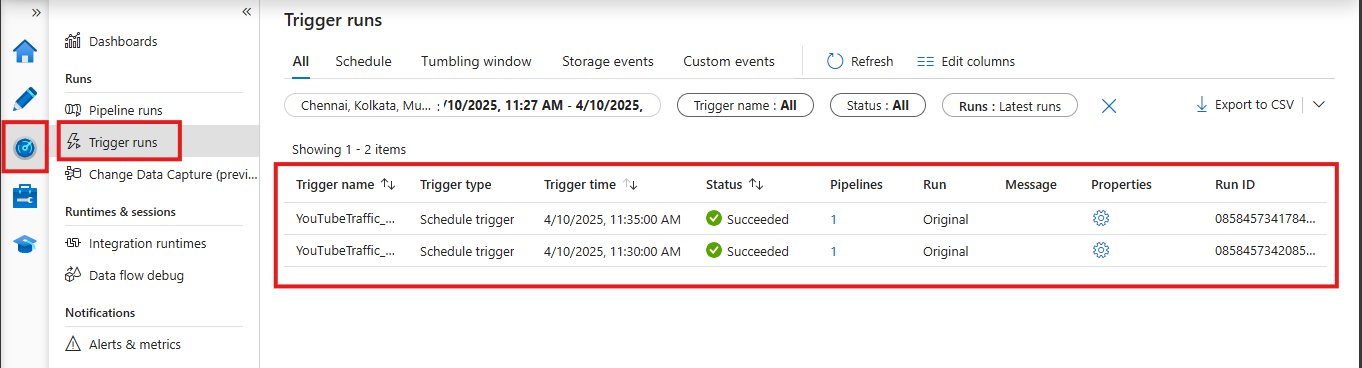


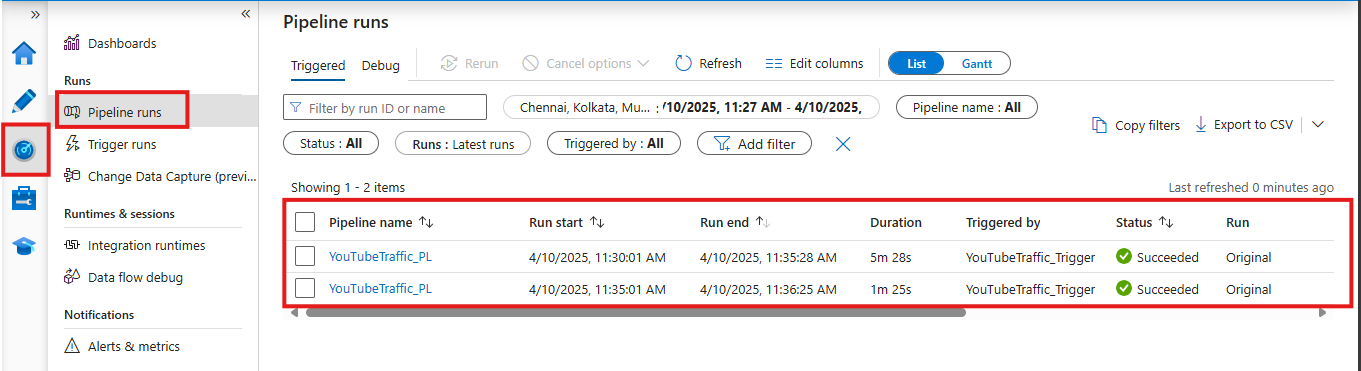
Step5: Create a trigger – to schedule the pipeline to run every 5min.





Step6: Monitor the Trigger runs, and Pipeline runs.

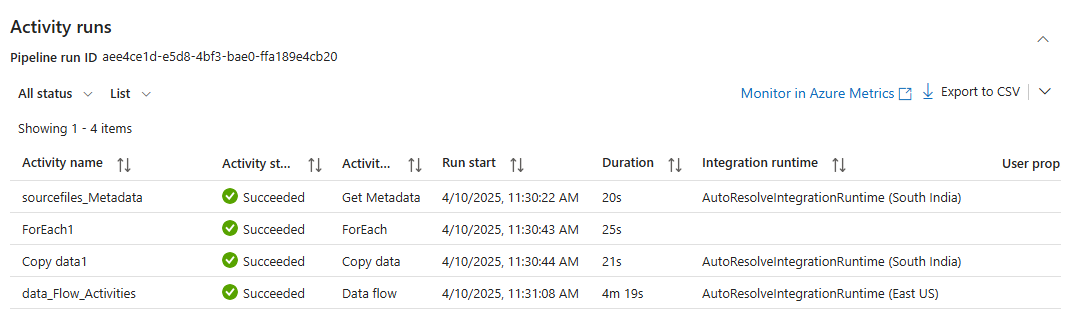




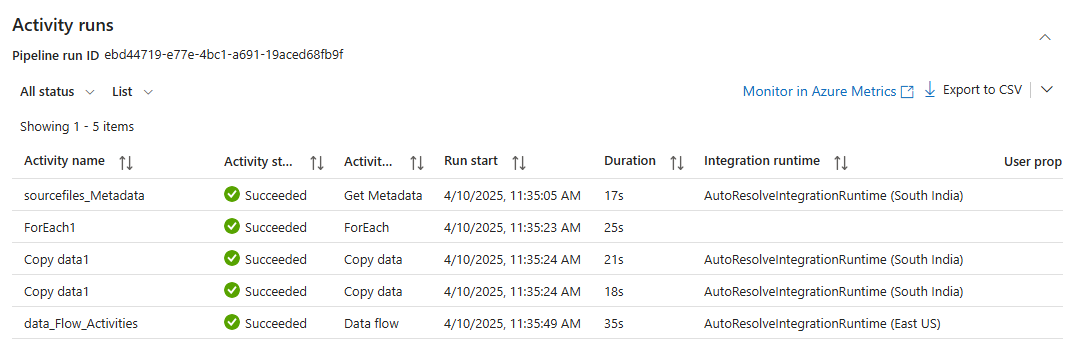
2 times, the pipeline ran.  
In the first run, there was only 1 file i.e., Dataset1.xls

In the second run, there was two files i.e., Dataset1.xls & Dataset2.xls. First both files are moved/copied to staging area. Then, both are files are merged and filtered using data flow activity and sink in transformed area.

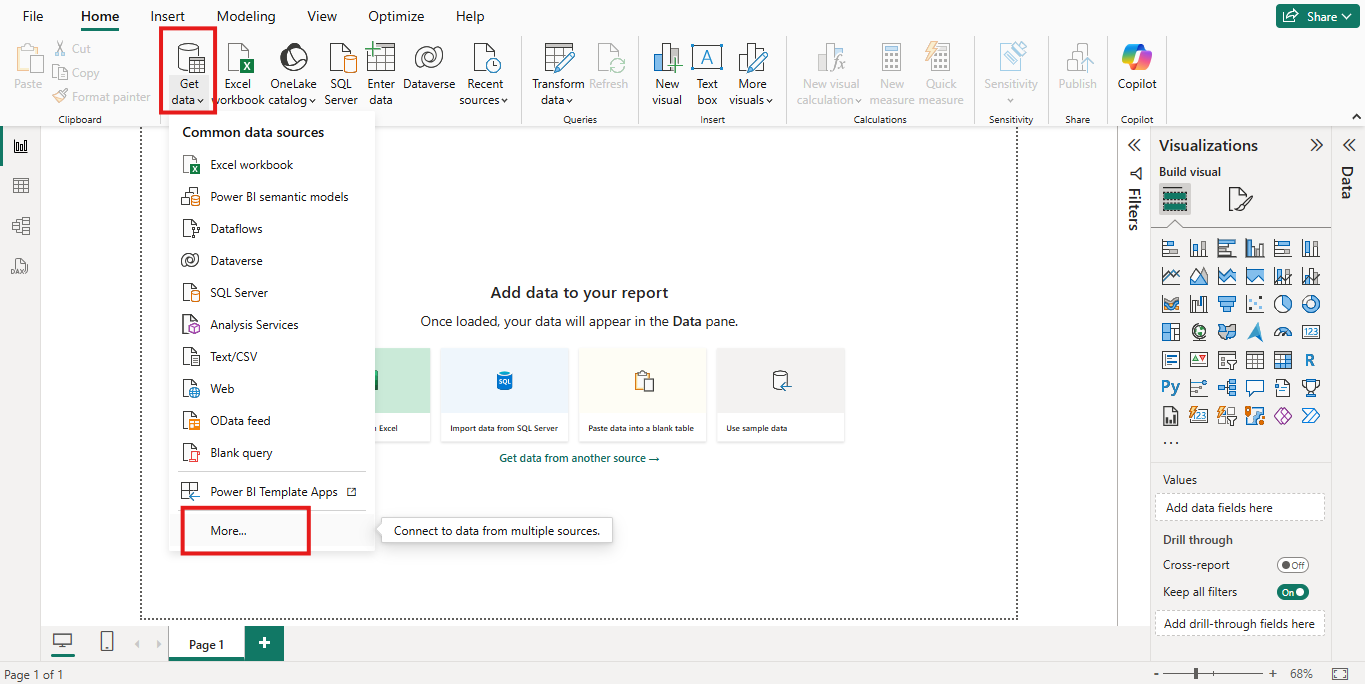
Run1:

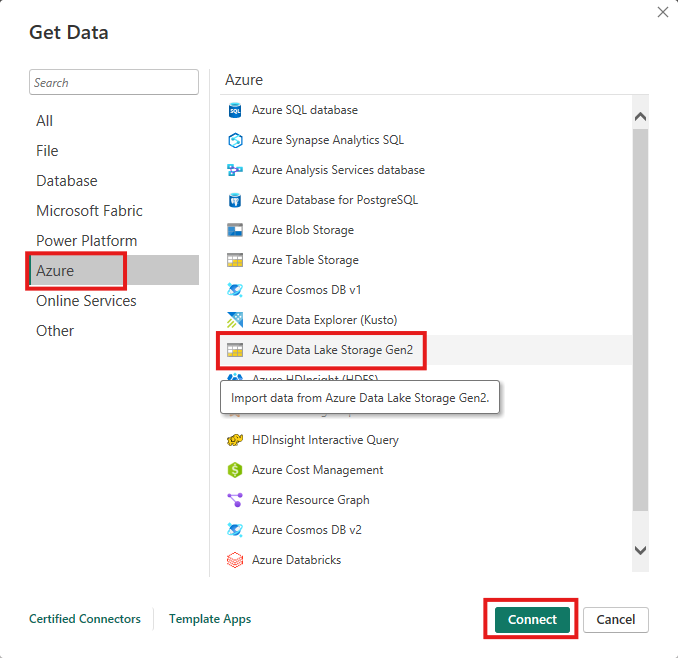


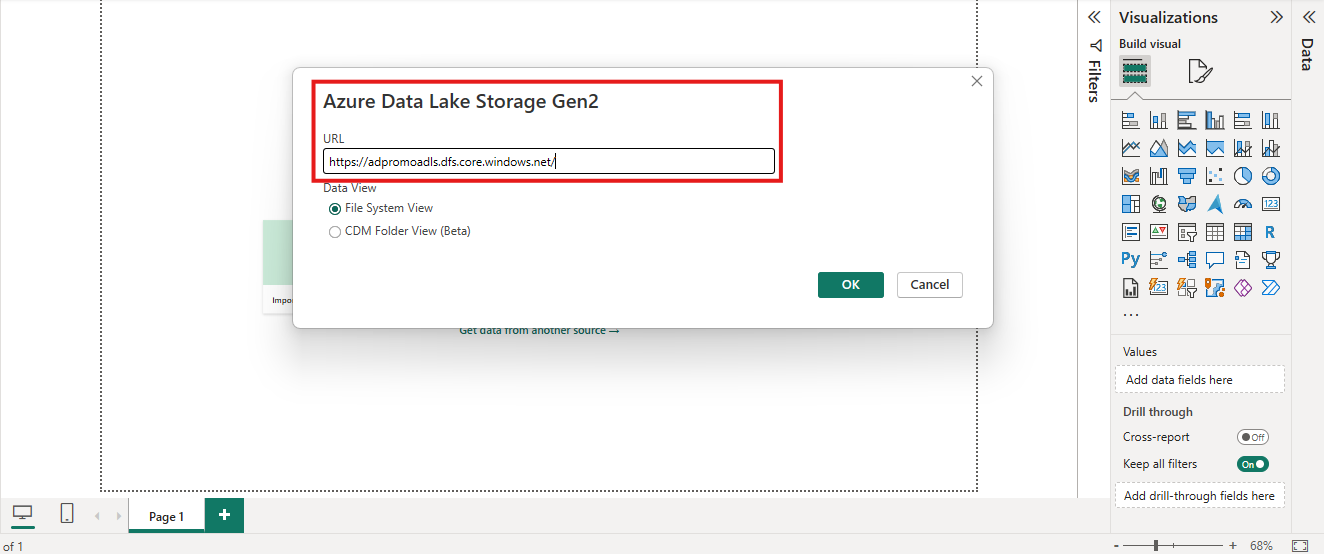
Run2:



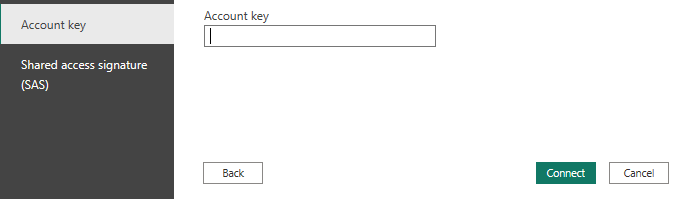
Step7: Connect PowerBI with ADLS Gen2 for visualization.



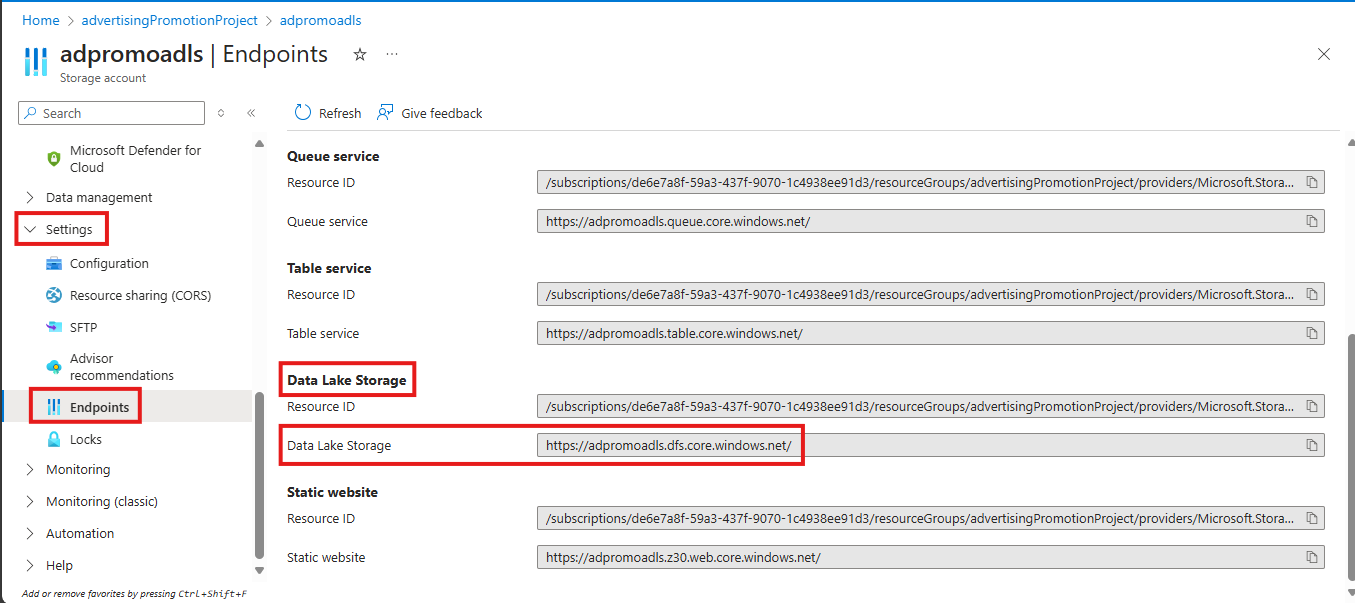




Provide storage account URL, as our data for visualization is residing in ADLS.  
We can get the URL from storage account >> dashboard >> Settings >> Endpoints >> Data Lake Storage.

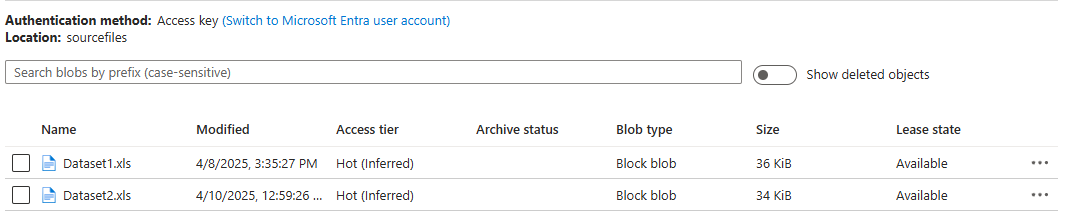


storage account >> dashboard >> Security + networking >> Access keys >> key1.

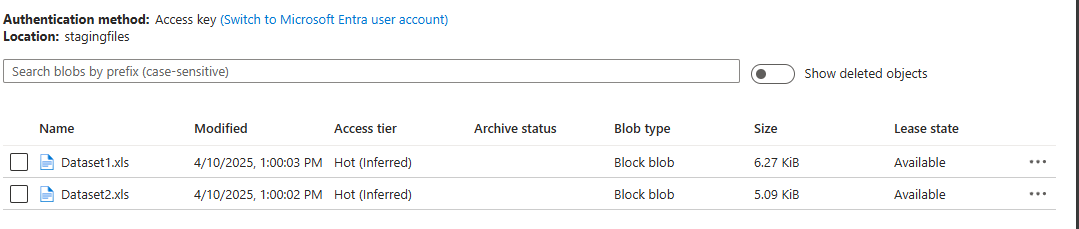


Outputs:

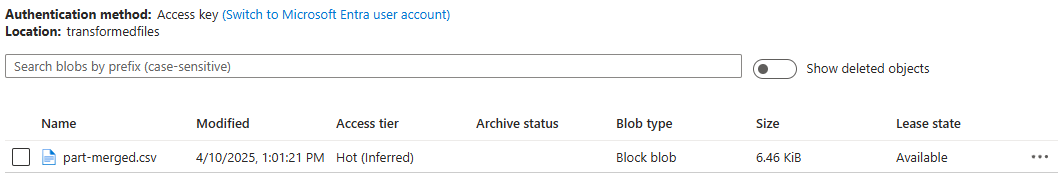
ADLS Source:



ADLS Staging:

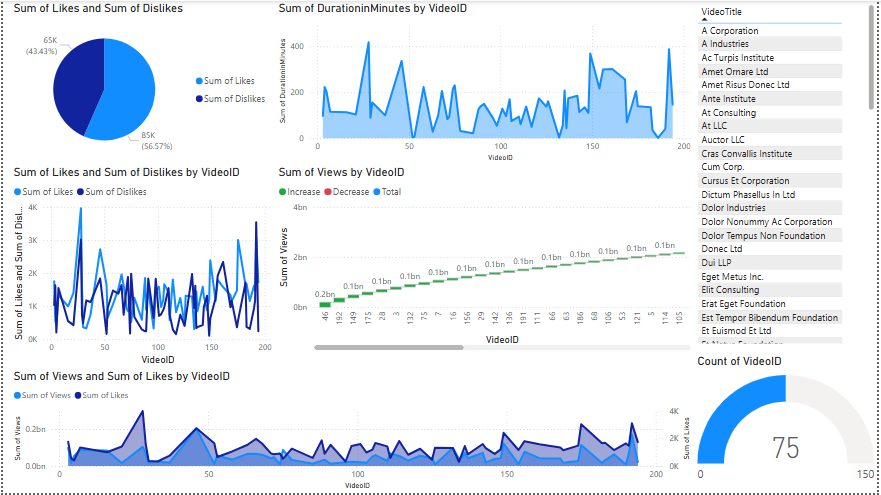


ADLS Transformed:



Power BI Outputs:

Dataset1



Dataset1 & Dataset2.

