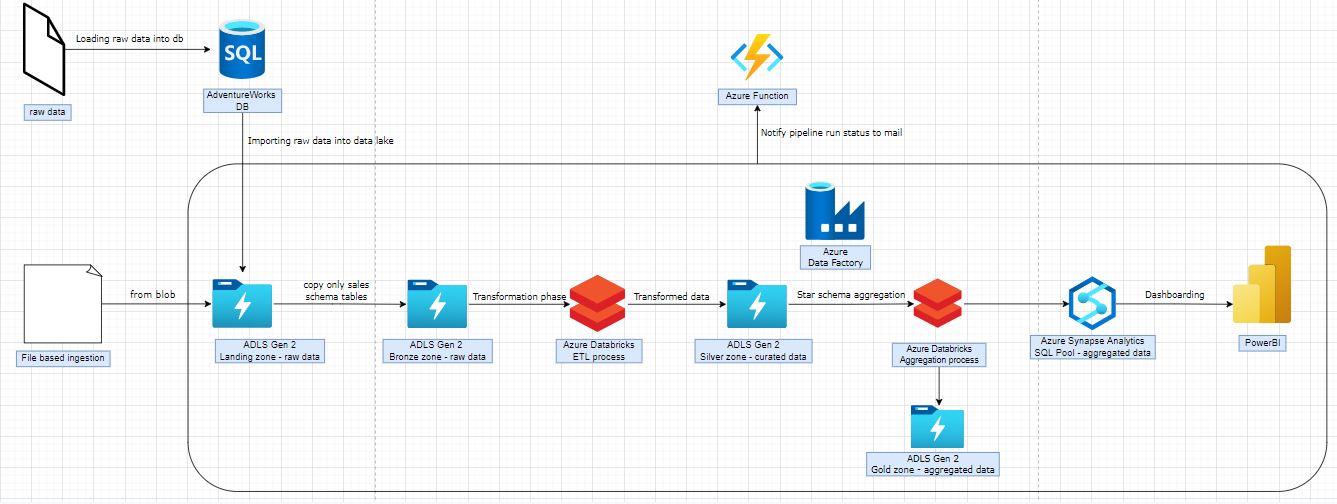
**Business Requirement:**

A company wants to use the Azure cloud platform to store big data in different formats in a cost-effective way. This data will be stored in different formats which need to be consumed by business users for multiple purposes like descriptive analytics. This pipeline should pick data from a source and ingest it into the data lake. The data lake should contain a landing zone, bronze zone (raw data), silver zone (curated data), and gold zone (aggregated data). The business user should be able to use this data using simple & popular tools effectively & easily like Power BI. As a Data Engineer, you need to design and implement an end-to-end solution that can be scalable, cost-effective, and can be maintained easily.

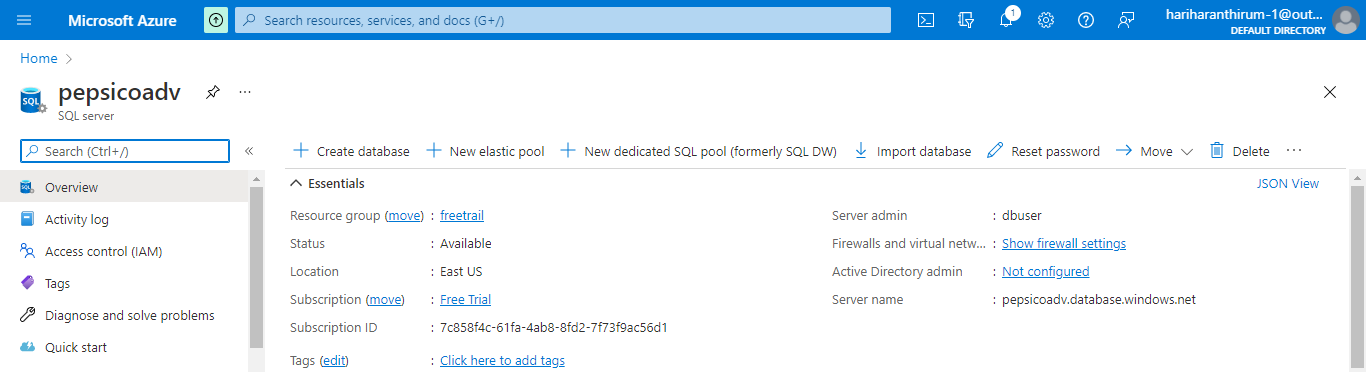
**Milestone 1:**

Understand the business requirement and create a high-level architecture.

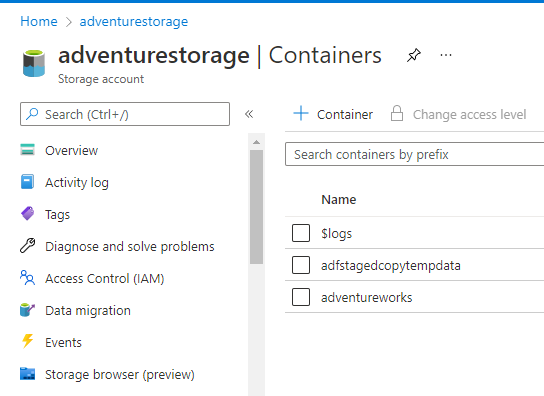
****

**Prerequisites:**

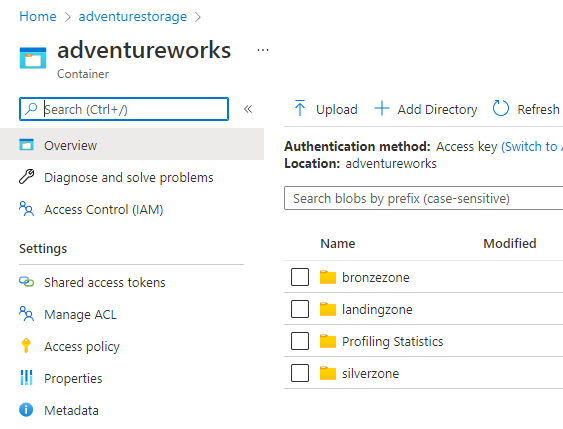
1. Create a SQL server for further database creations and usage:



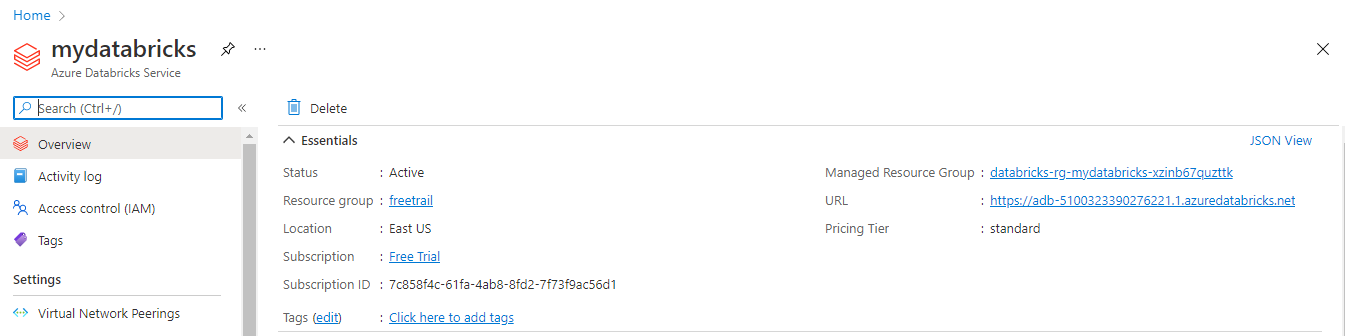
1. Create a Data Lake Gen 2 Storage called “adventurestorage”, then create a container named “adventureworks” to store the processed data after cleaning and aggregating as shown below:



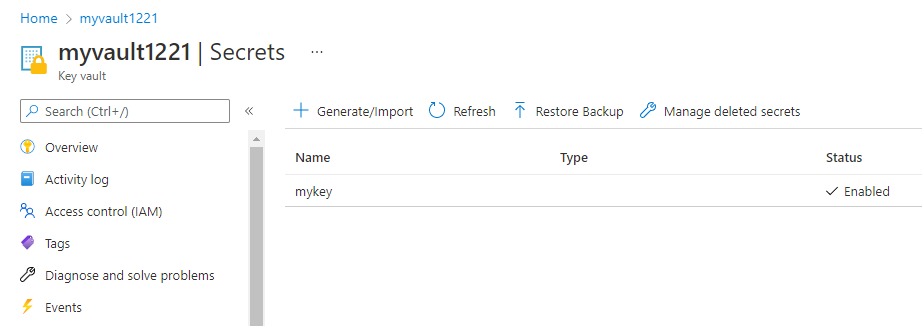
1. Create 3 different directories representing each zone to store the data and create another folder named “Profiling Statistics” to store data profile.



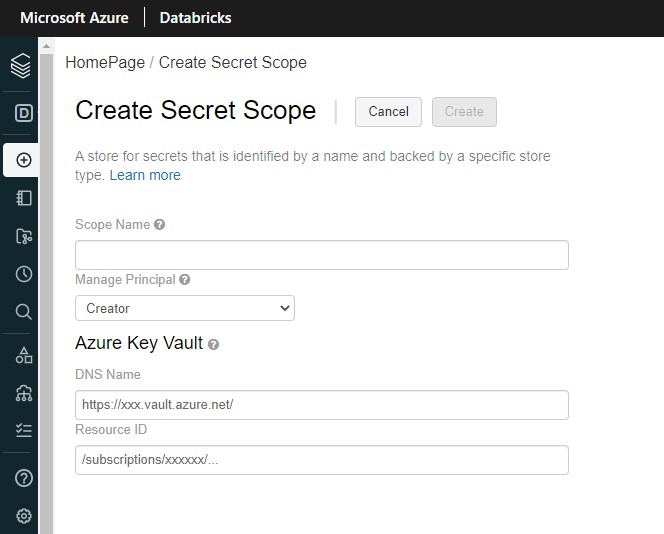
1. Create a Databricks called “mydatabricks” as shown below for further cluster and notebook usage:



1. Creating a connection to data bricks:
2. Create a key vault called “myvault” and create a secret with name “mykey” using ADLS Gen 2 access key as value:



1. Create a Secret Scope by adding [#secrets/createScope](https://adb-3198008930614243.3.azuredatabricks.net/#secrets/createScope) at the end of the databricks URL. Use the “Vault URI” of key vault in “DNS Name” and use the “Resource ID” of key vault in “Resource ID” (from key vault properties) and create a secret scope as below:



1. Use the scope name as “myscope” while configuring spark and key vault “mykey” name as secret.

**Milestone 2:**

Create AdventureWorksLT database and create a data dictionary

1. We going to create a database using sample database through azure database creation services as shown below:

Graphical user interface, text, application, email

Description automatically generated

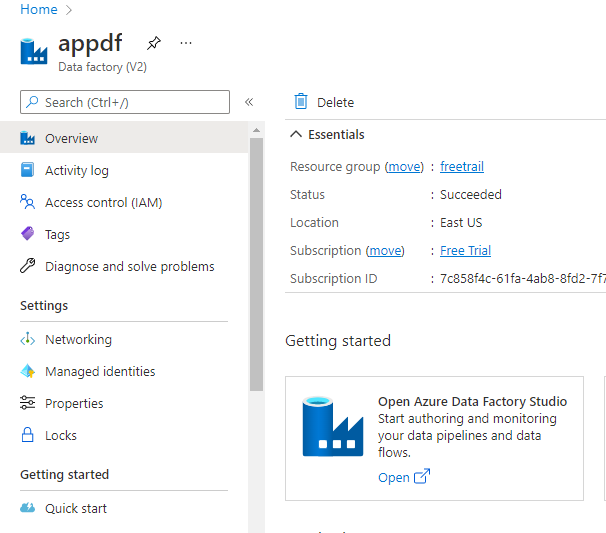
1. Once the database is created use [this](https://docs.google.com/document/d/1RdFGkVZq-N3BNT5AoSWCtwdm8deHicLjXwBZD5Zp_iU/edit?usp=sharing) query to create a [DATA DICTIONARY](https://docs.google.com/spreadsheets/d/1M2dMuIj0d--DtFBFgUJHZTgaf9O7wgbd/edit?usp=sharing&ouid=107919227372254138313&rtpof=true&sd=true)

**Milestone 3:**

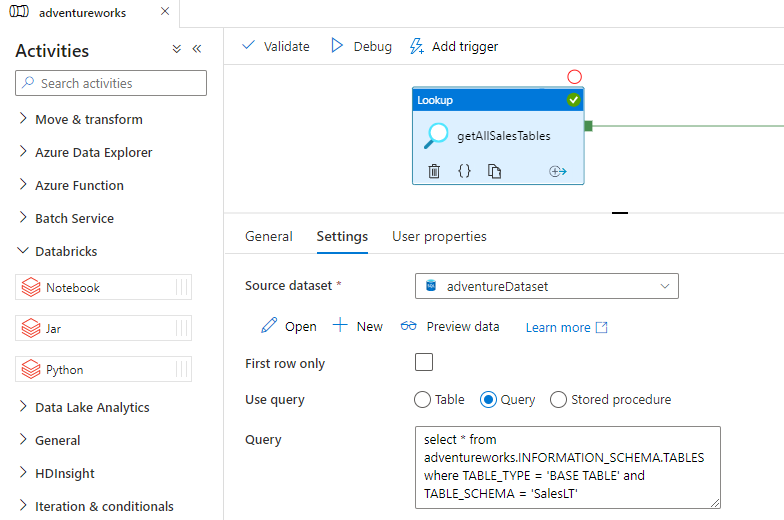
Use ADF to copy data from Azure SQL to Landing Zone to bronze zone

**Data Pipeline:**

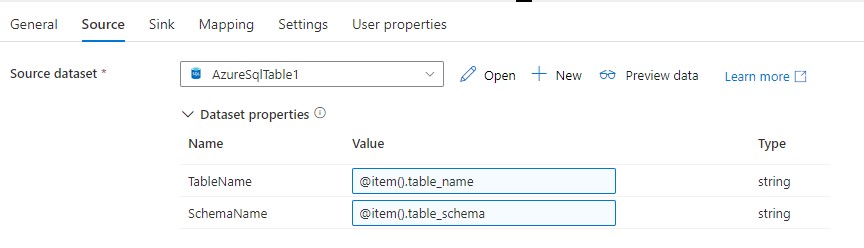
1. Create a Data Factories called “appdf” and click on “open Azure Data Factory Studio” as shown below:



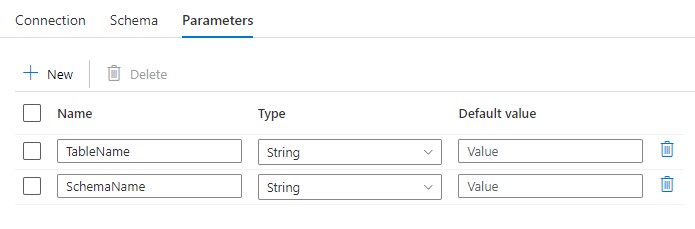
1. Click on “Author” and create a pipeline named “adventureworks”
2. Drag and drop lookup for creating data lookup up for adventure works database and give query as shown in picture. (Note: do not select any tables while creating dataset)



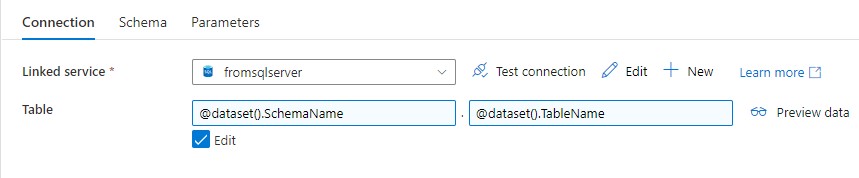
1. Drag and drop “for each” and connect it to lookup and use below settings
   1. Generate a source dataset “azuresqltable1” for foreach source and use below settings. (Note: do not select any tables while creating dataset)



* 1. Go to parameters and add two parameters as below

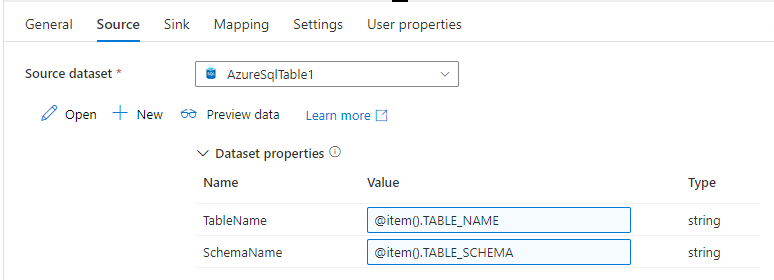


* 1. Click “open” and enable the edit option add using add dynamic content enter the table name as below

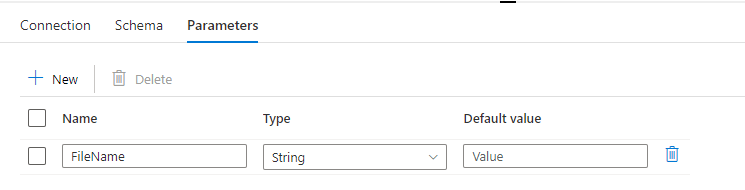


* 1. Now go back to the foreach item and add “TableName” and “SchemaName” using add dynamic content option.

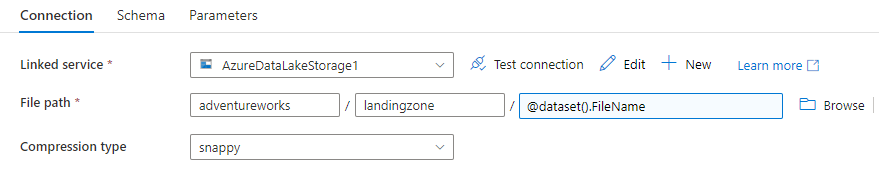
(Note: select foreach item and append “.TABLE\_NAME” for TableName and “.TABLE\_SCHEMA” for SchemaName respectively)



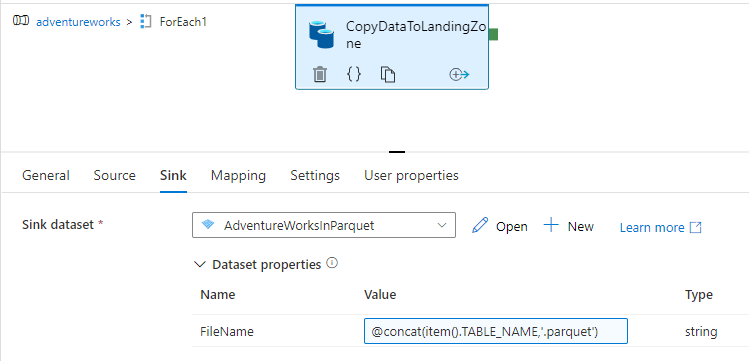
* 1. In sink, create a new dataset with format as “Parquet” and specify the landing zone directory for the location. Then go to the dataset and add parameter named “FileName”



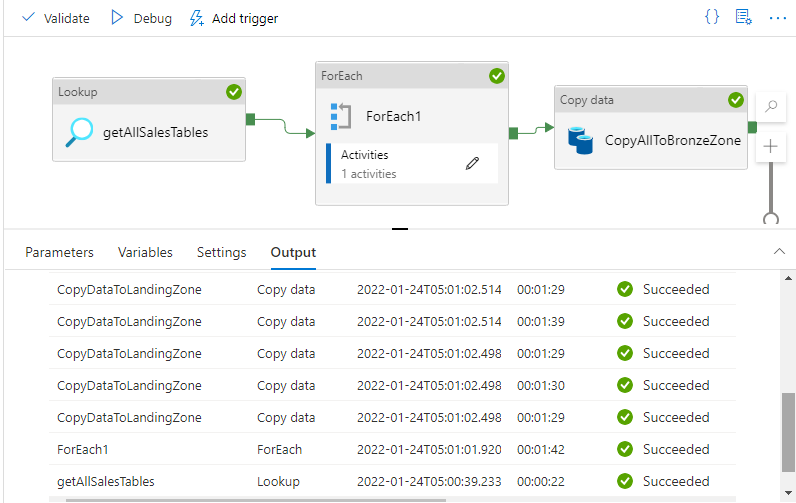
* 1. In connections use dynamic content and add the file name as below



* 1. Go back to the foreach item and in sink give the File Name as below to not use schema name for the parquet file name as we’re taking only from a single schema



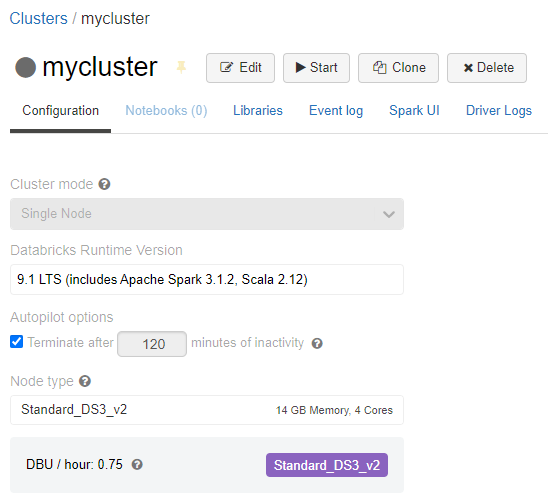
1. Add another copy data outside the foreach loop which will copy data from landing zone to bronze zone. (Note: use existing linked service and make sure the source and sink are in correct directory, use parquet for the destination dataset also)
2. After completing the above 5 steps, validate the pipeline. On successful validation, debug the pipeline. Once done, the results will be shown as below



**Milestone 4:**

Transformations using Azure Databricks

1. Create a Cluster using Cluster mode as “Single Node” and click on create as shown below and click on start cluster:



1. Using the cluster created, we can start writing the code for transformation. Notebook link is given [here](https://drive.google.com/file/d/1YQHCwkNwDNaa_OzbIus2eYRXfk33CYeK/view?usp=sharing) for the reference.

**Milestone 5:**

Make the pipeline **configuration driven**

1. A Configuration is written in json format which will hold all necessary parameters as a key value pair. Check the file [here](https://drive.google.com/file/d/1ezwnmou0DxYvaM6h7uFJ5SLAl2LeThHe/view?usp=sharing)
2. Notebook code is modified in a way that reading, writing, null handling and dropping the columns will be done dynamically.
3. Registered delta tables inserted some records and updated existing records.
4. Data profile will be done and stored each time the code is run with a timestamp.

**Milestone 6:**

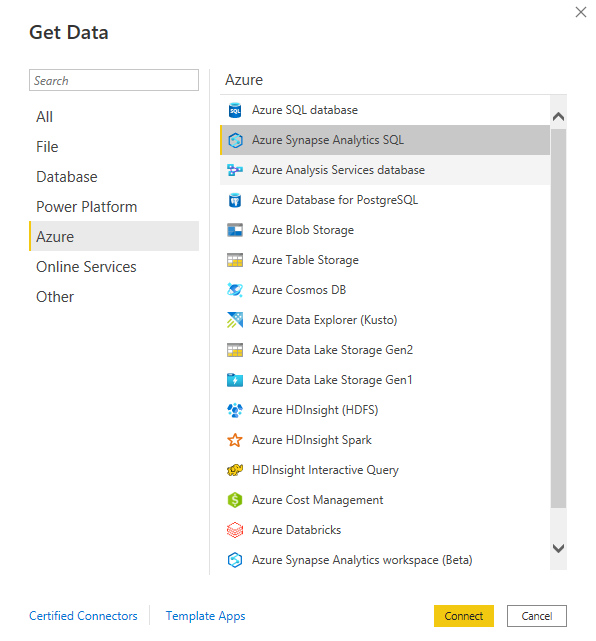
Decide a fact table and dimension tables for star schema creation

1. Create a synapse workspace and create a sql pool.
2. Create tables which is used for star schema. Check the sql table creation query [here](https://github.com/tigerrepository/incubation_lab/blob/main/Hariharan_Azure/DataFlow/milestone%206%20query%20-%20Synapse%20SQL%20Pool.sql).
3. Create source datasets from silver zone and join them based on columns.
4. In our case, we have one fact table and two dimension tables
   1. Fact table – **SalesFact**using *SalesOrderHeader* left join *SalesOrderDetail* tables based on SalesOrderId.
   2. Dimension table(1) – **CustomerDim** using *CustomerAddress* left join Address based on AddressId left join *Customer* left join based on CustomerId.
   3. Dimension table(2) – **ProductDim** using *Product* left join *ProductCategory.*
5. Result datasets should be moved into Synapse sql tables and gold zone.

**Milestone 7:**

Import tables from synapse pool into PowerBI for dashboarding

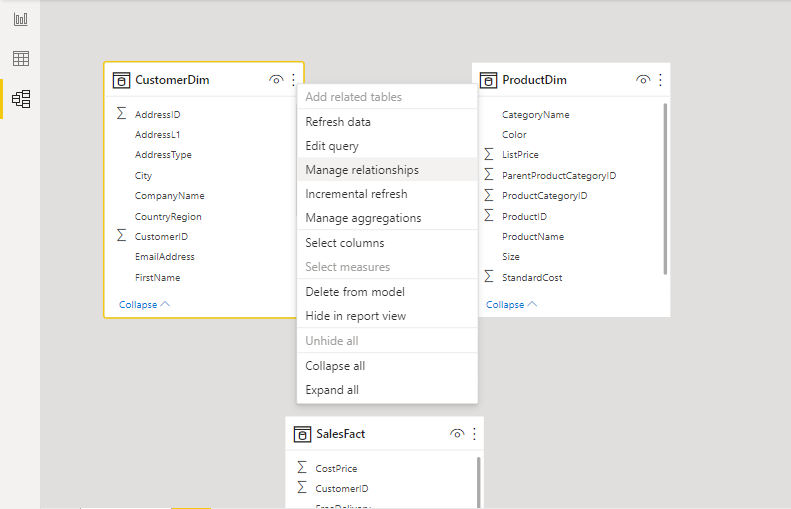
1. Download PowerBI desktop app and open it.
2. Click Get Data -> Azure -> Azure Synapse Analytics SQL and connect it.



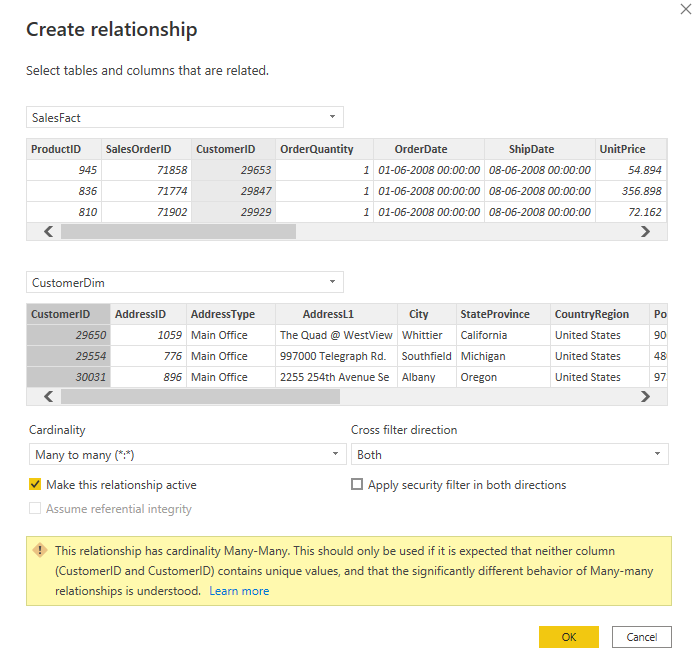
1. Enter server name and select data connectivity mode as import and click ok.



1. Go to model and establish relation between tables.
   1. Click options of any tables and click manage relationship



* 1. Click new and create relationships like below.

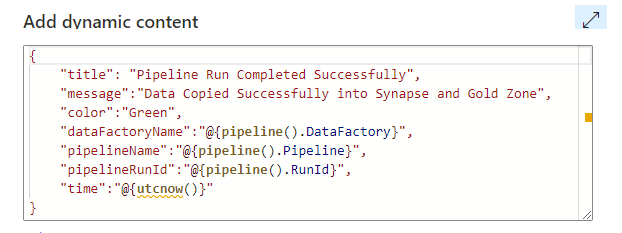


* 1. Once relationship between tables is done, we can start dashboarding. Click [here](https://github.com/tigerrepository/incubation_lab/tree/main/Hariharan_Azure/PowerBI) to view my dashboard

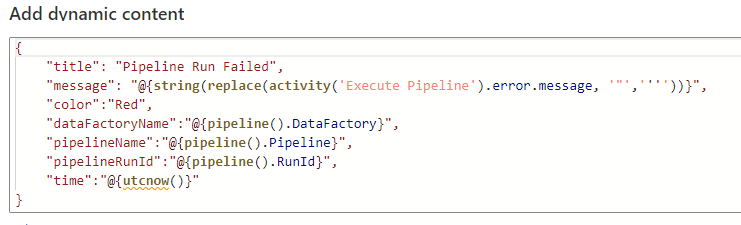
**Milestone 8:**

Sending Email notification on Success/Failure

1. Using web activity POST method, send the contents required in mail as a JSON like below
2. on success:



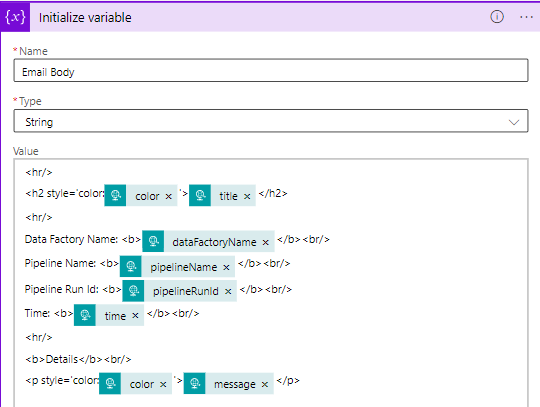
1. on failure:



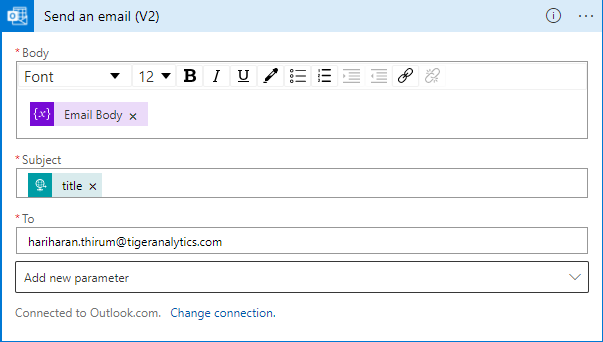
1. This activity can be done in many ways, here we used logic app and function app
   1. Logic app
      1. Create a logic app and add a HTTP trigger
      2. Enter sample payload like below



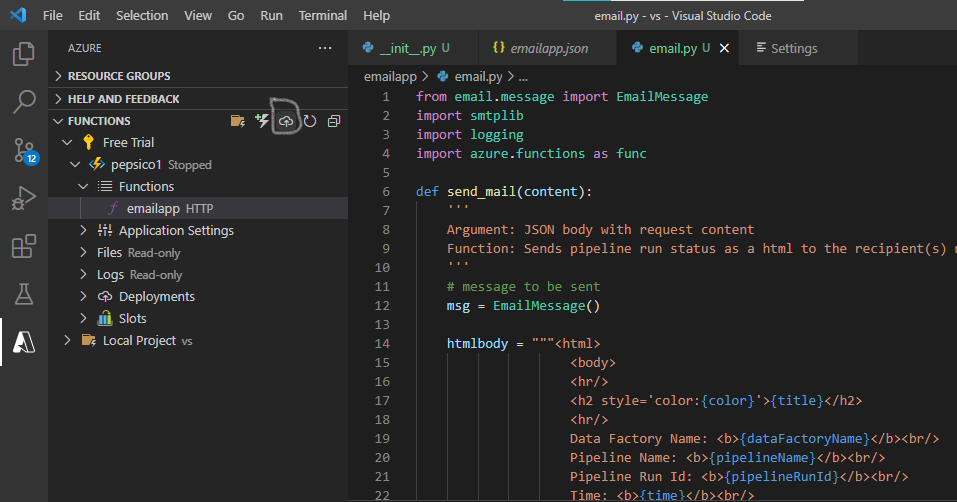
* + 1. Initialize a variable to use it in the mail body as HTML



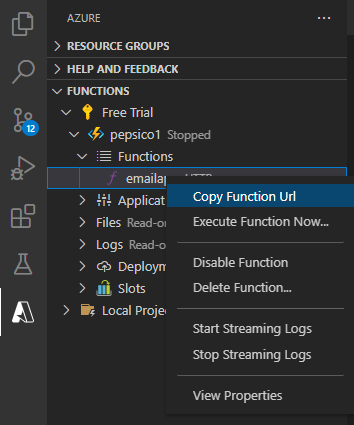
* + 1. Add another activity from **outlook.com**, use send an email(v2) from the actions tab and sign in your outlook account.
    2. Fill the values using dynamic contents as shown below



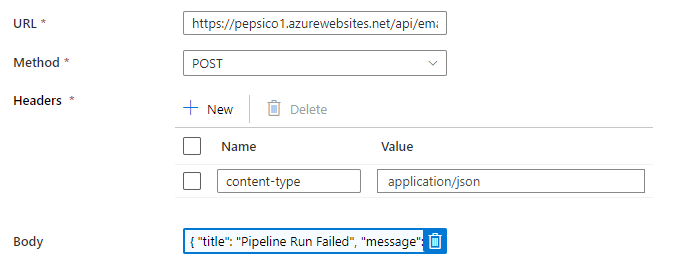
* + 1. Once done, save and copy the URL to use in ADF web activity
  1. Function App
     1. Create a function app with runtime stack as *python.*
     2. Download and install vscode -> Install Azure Function, Python from vscode extensions.
     3. Sign-in your azure account
     4. Create a function with HTTP trigger and use the code given in the [link](https://github.com/tigerrepository/incubation_lab/tree/main/Hariharan_Azure/Code/email%20(function%20app)).
     5. Once done, deploy the code by clicking the highlighted button



* + 1. Copy the function URL by right clicking the function like below

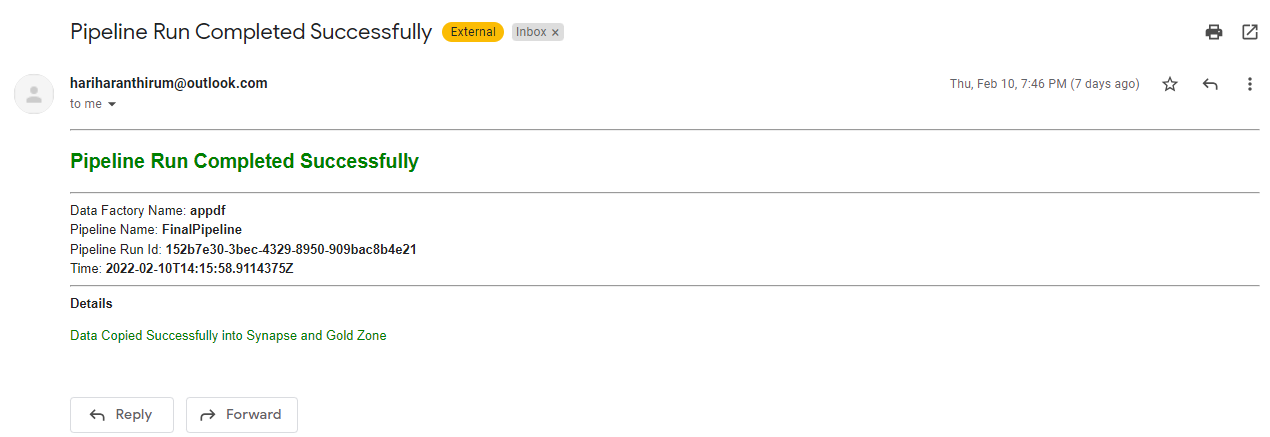


1. Paste the URL copied in ADF web activity and give content-type header like below screenshot

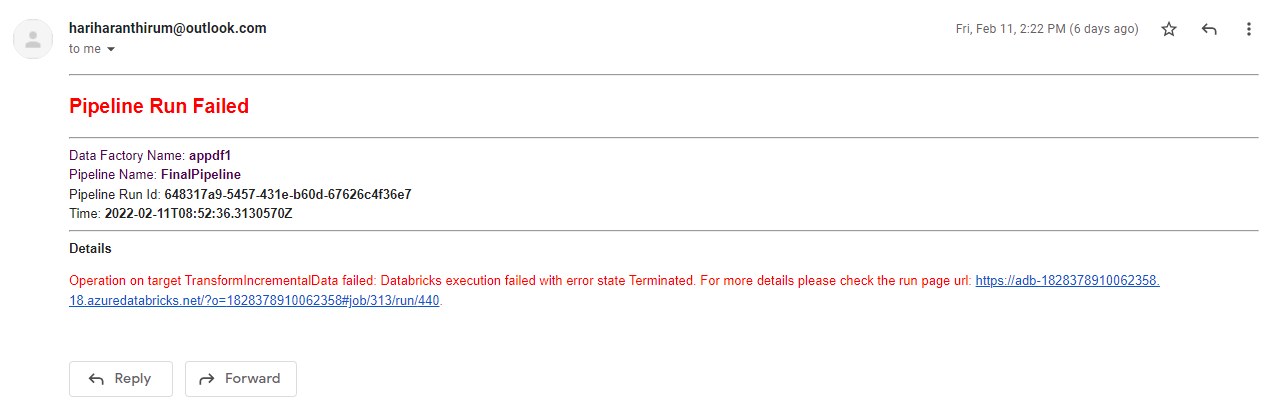


**Sample Mail Screenshot:**

On Success:



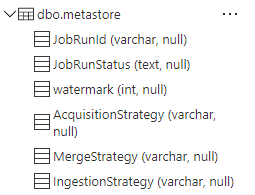
On Failure:



**Milestone 9:**

Design and develop a **SQL Metastore**

1. Create a table named ‘Metastore’ with columns as below



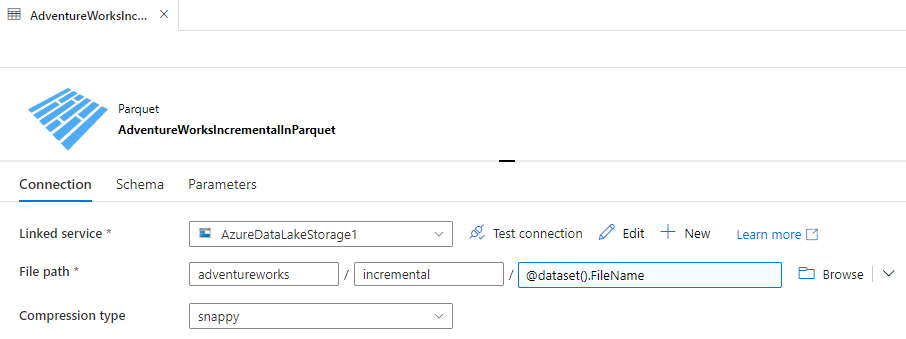
1. Insert a record only with watermark (if applicable), Acquisition Strategy and Merge Strategy.
2. Now if we run the pipeline, we have a lookup activity which will run the below query to determine whether the acquisition strategy is incremental or full



1. Using the result in an if condition activity, we copy the data either incremental or full.



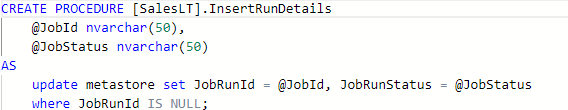
* 1. If incremental, create a dataset which points to a folder named **incremental**



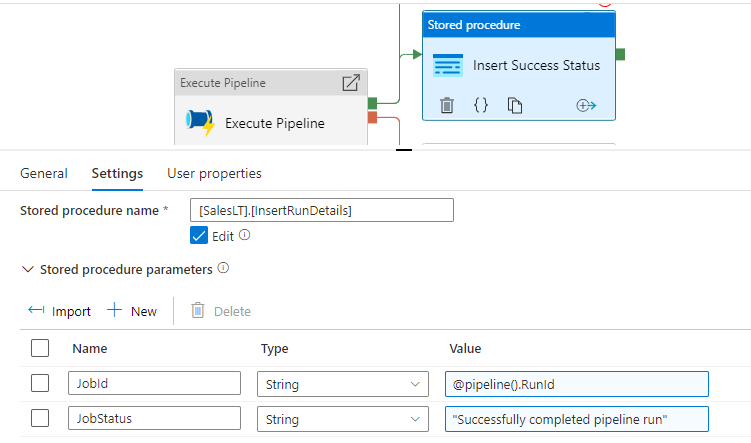
* 1. data will be stored in incremental folder and it will be appended to landing and bronze zones using databricks [notebook](https://github.com/tigerrepository/incubation_lab/blob/main/Hariharan_Azure/Code/M3%20(incremental)/Landing%26Bronze_SalesLT.ipynb).
  2. Else, existing copy data will load the data completely.

1. Again, using the same if condition activity, transformation will occur using databricks [notebook](https://github.com/tigerrepository/incubation_lab/tree/main/Hariharan_Azure/Code/M4%2C%20M5).
   1. If incremental, only the data in incremental folder will undergo transformation. Once done, it will be appended to the existing bronze zone data and data profiling will be done overall.
   2. Else, existing transformation will occur transforming full data.
2. After inserting data into gold zone and synapse sql pool, we will update the job run id and job run status in the metastore table using a stored procedure.

Stored Procedure Definition



* 1. On success

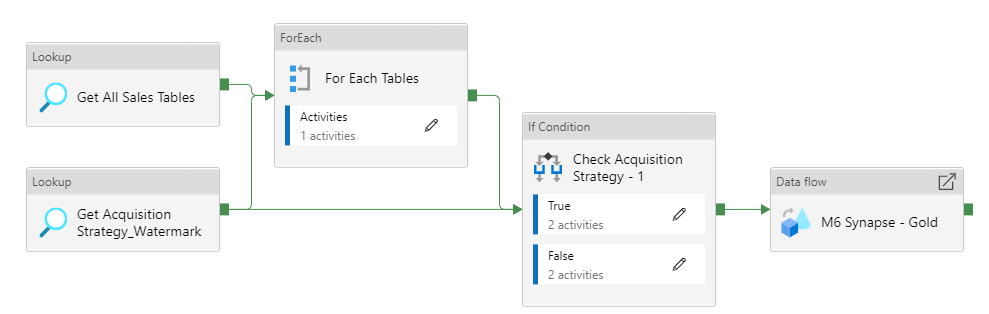


* 1. On Failure

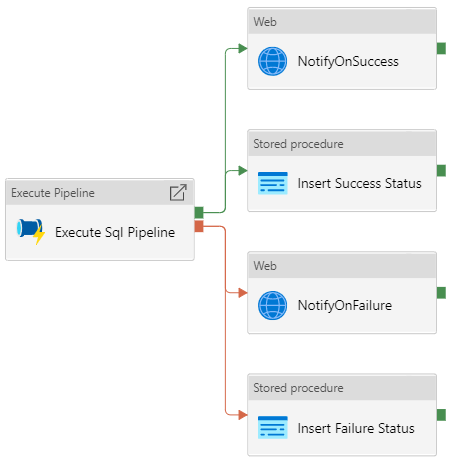




**Core Pipeline after Milestone 9:**



**Pipeline for Execution:**



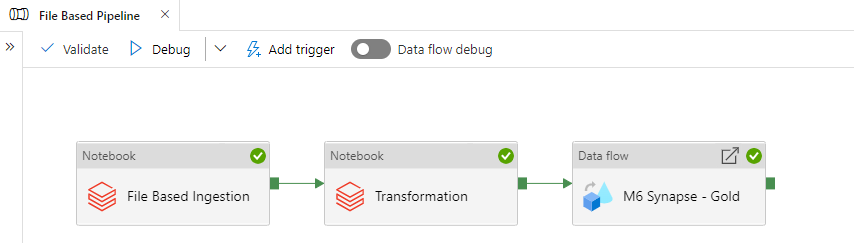
**Milestone 10:**

File Based Ingestion

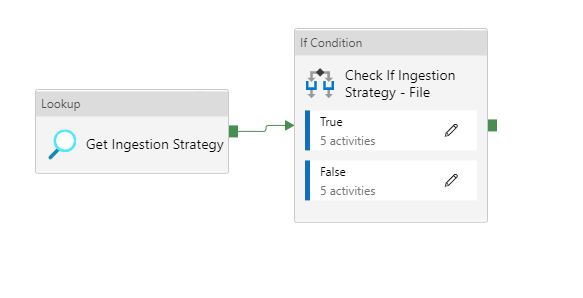
1. Using faker package, created a sample data for all 10 tables in SalesLT schema. Notebook [link](https://github.com/tigerrepository/incubation_lab/blob/main/Hariharan_Azure/Code/M10/File%20Based%20Ingestion.ipynb).
2. Created a new pipeline “File Based Ingestion” and added sample data creation notebook as first step. It will load the data into both Landing and Bronze Zones.
3. For next step, used the same notebook “Transformation” which is done for Milestone 5. It will load the data into silver zone. Notebook [link](https://github.com/tigerrepository/incubation_lab/blob/main/Hariharan_Azure/Code/M4%2C%20M5/SalesLT_transformation%20(full).ipynb).
4. Last step is done by Cloning the existing data flow which is created for Milestone 6. It will load the data into Star schema tables and gold zone.
5. Insert a record only Acquisition Strategy, Merge Strategy and Ingestion Strategy as ‘file’.
6. Added another base parameter in transformation notebook to determine where it is file-based ingestion or not.



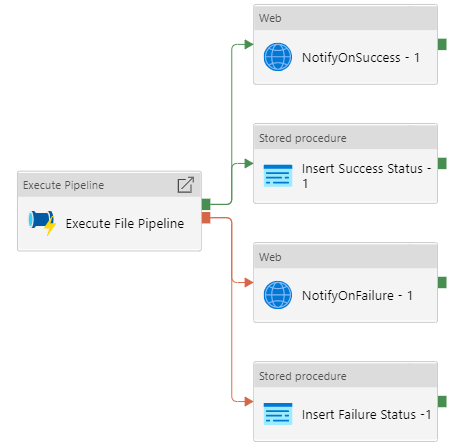
**Core Pipeline for Milestone 10:**

****

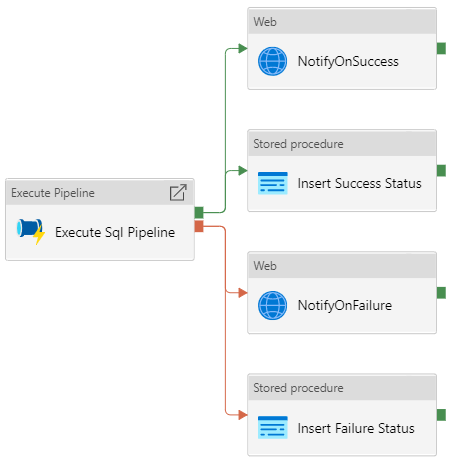
**Pipeline for Execution:**



* **True Activity:**



* **False Activity:**

****