# End-To-End Data Engineering Project using Microsoft Azure and Azure Databricks

# Business Problem

## Context:

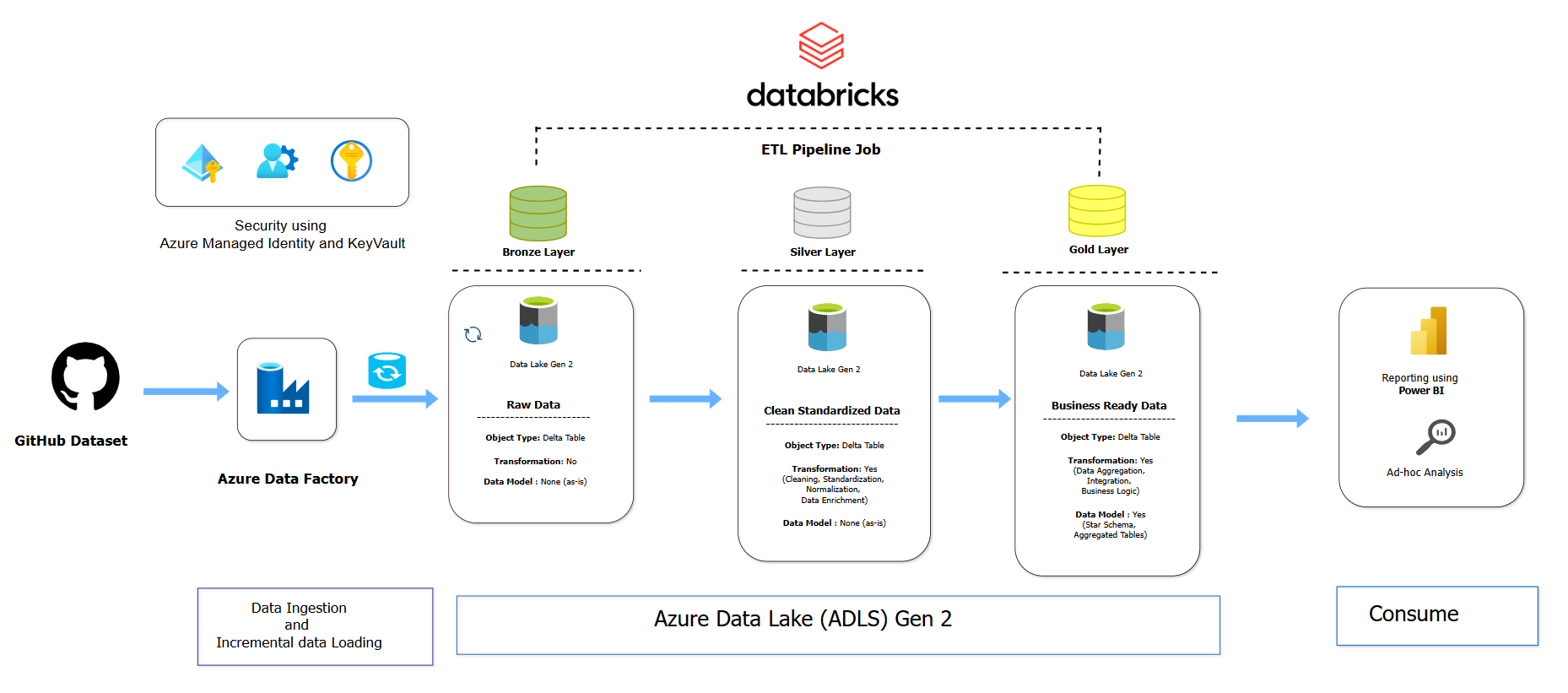
A retail company (or a similar enterprise in this case study) operates in multiple regions with diverse product lines and a network of resellers. Sales are driven by a distributed salesforce, and the company sets quarterly/annual sales targets at different levels (by product, region, or salesperson). However, the company faces challenges in tracking performance, identifying underperforming regions/resellers, and aligning sales strategy with business targets.

## Key Business Problems:

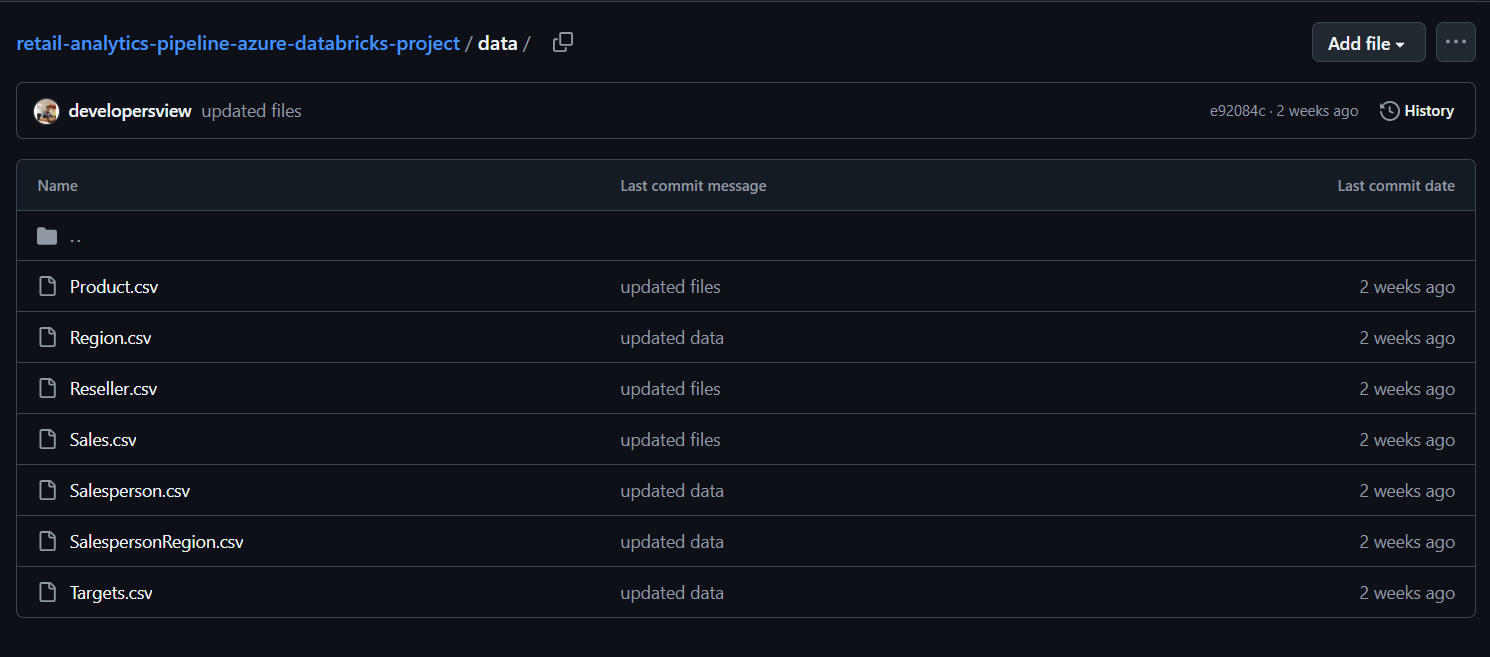
* Lack of unified view of sales across regions, products, and resellers.
* Difficulty in tracking sales performance vs. targets.
* Inefficient allocation of resources to underperforming areas.
* Limited forecasting and strategy alignment due to poor data insights.

## Proposed Data-Driven Solution:

* Ingest raw CSV datasets into **Azure Data Lake Gen2** using **Azure Data Factory**.
* Store raw, untransformed data in the **Bronze Layer** (Delta Tables) in Data Lake.
* Apply cleaning, standardization, and enrichment in the **Silver Layer**.
* Aggregate and apply business logic in the **Gold Layer** (Star Schema).
* Secure the pipeline with **Azure Managed Identity and Key Vault**.
* Deliver insights via **Power BI dashboard.**



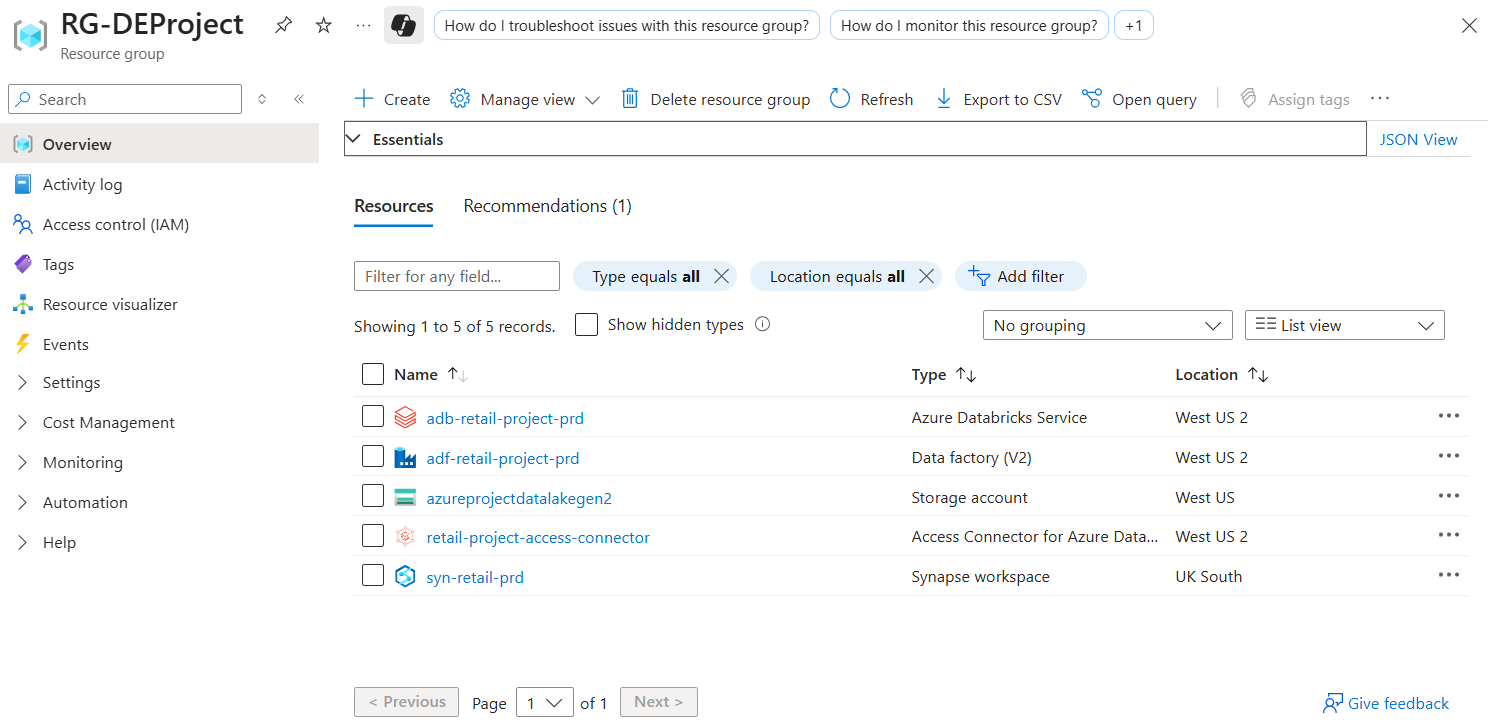
# GitHub Dataset



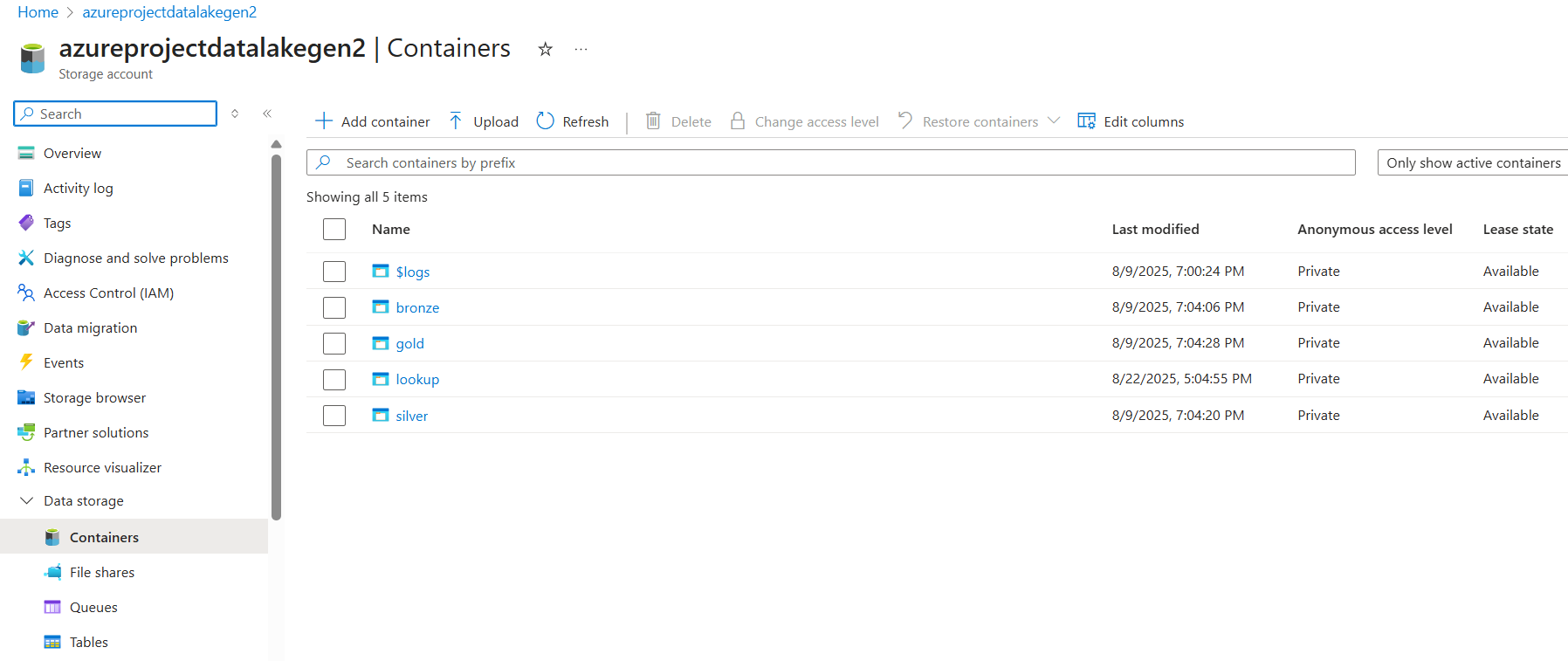
# Resource Group

Resources used:

1. Azure Databricks
2. Azure Data Factory (ADF)
3. Azure Data Lake Storage (ADLS) Gen 2
4. Azure Databricks Access Connector

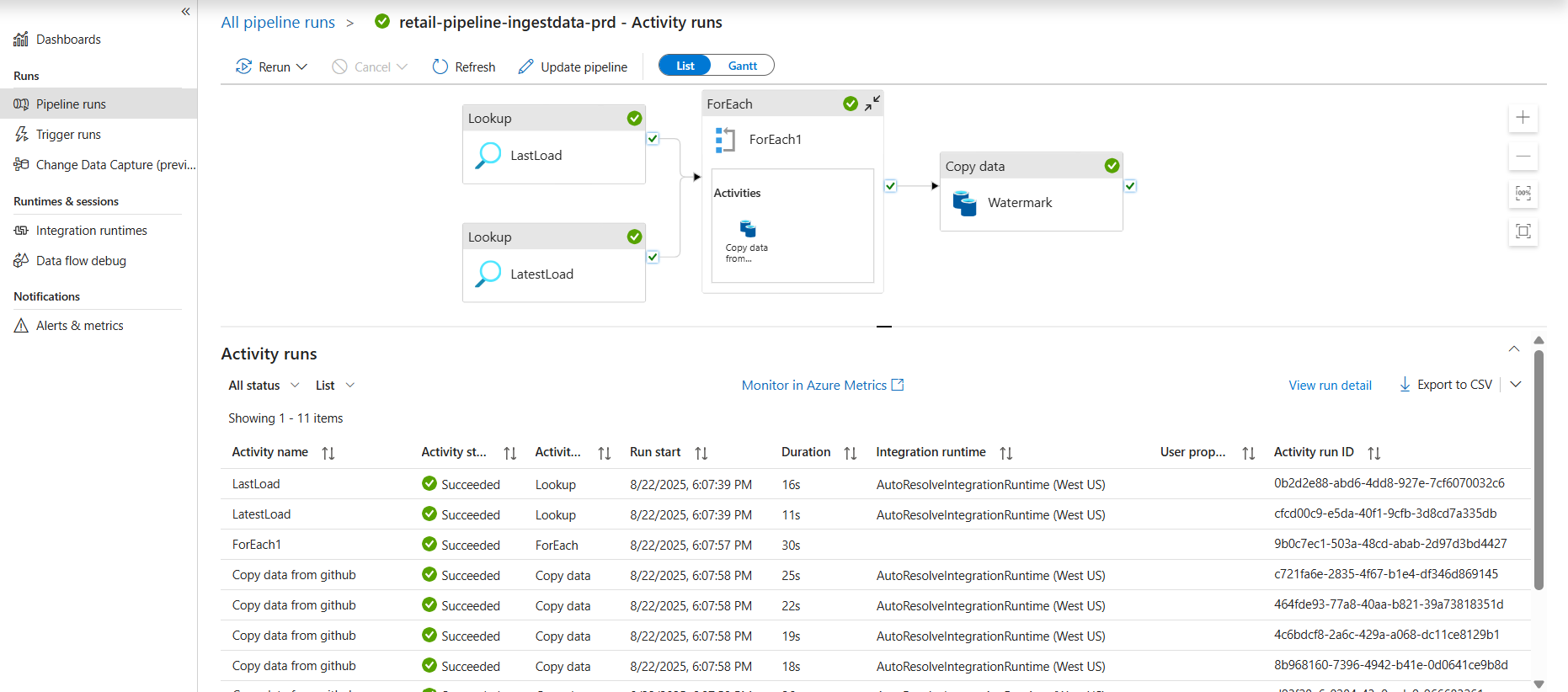


# Azure Data Lake Storage (ADLS) Gen 2



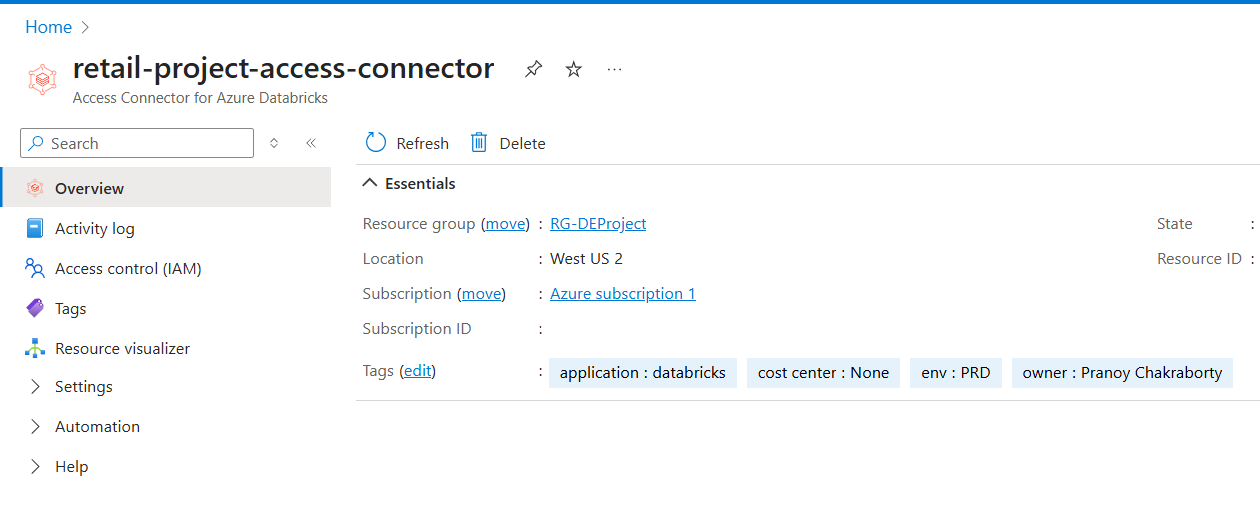
# Azure Data Factory Pipeline – Data Ingestion + Incremental Data Loading

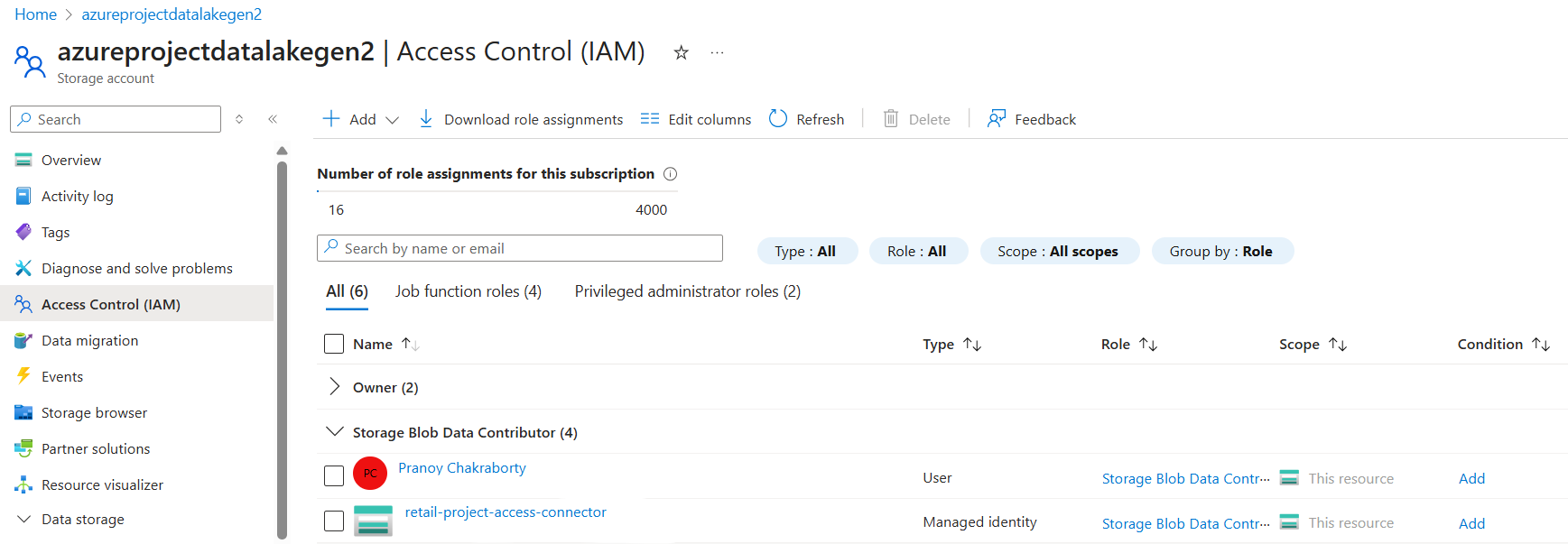
1. Use Lookup Activity to check the Last load value
2. Use another Lookup activity to check Latest Load value
3. Use Copy Activity nested in a ForEach Activity to copy data from GitHub or Any other source
4. Use Copy Activity to update the Last load value with Latest Load



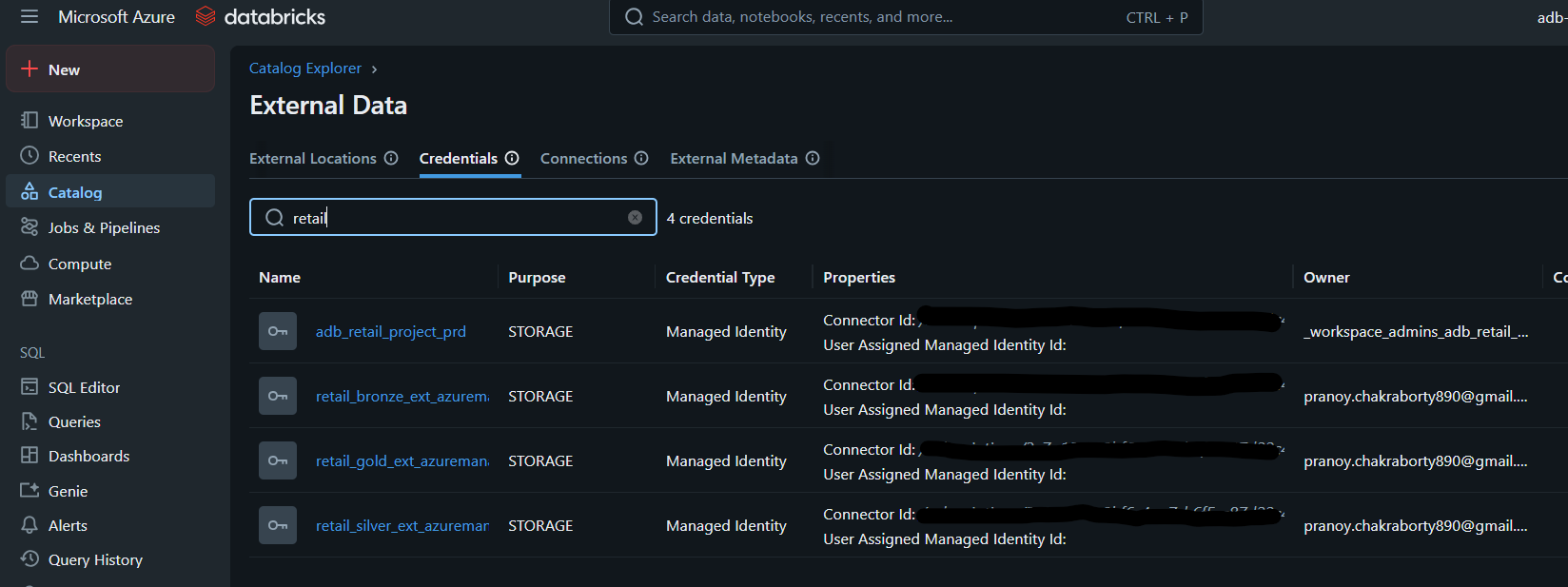
# Azure Databricks Access Connector

Azure Databricks Access Connector is required to access Azure Data Lake from Azure Databricks, without the access connector, Azure Databricks won’t be able read / write data from / to Azure Databricks

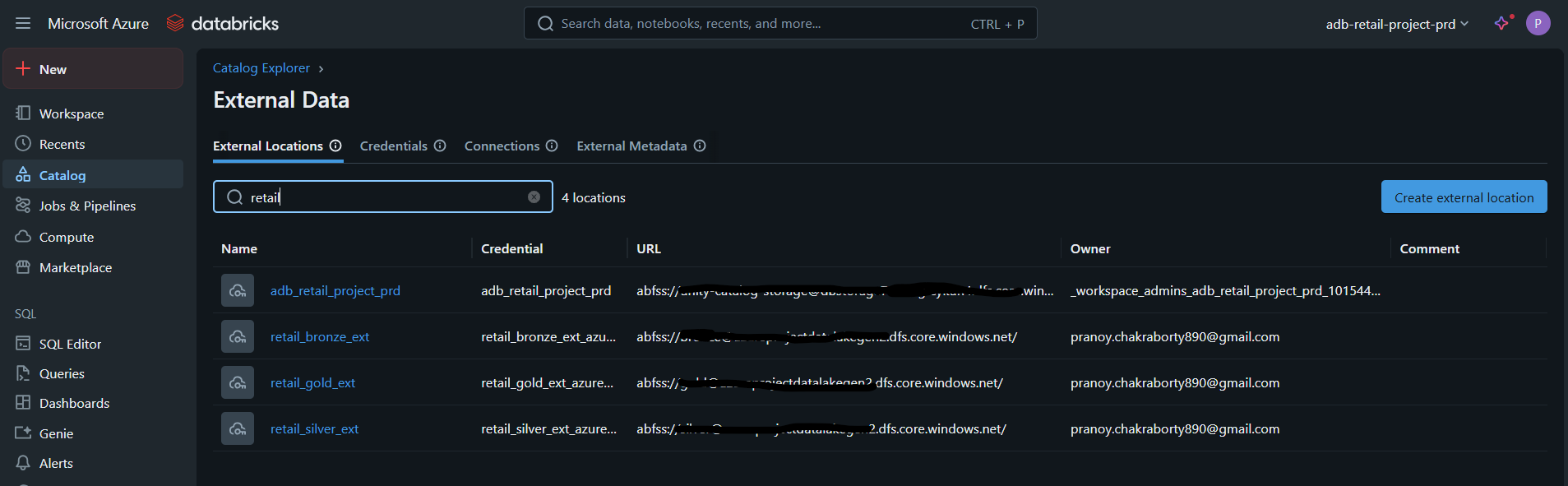




### Azure Databricks Credentials



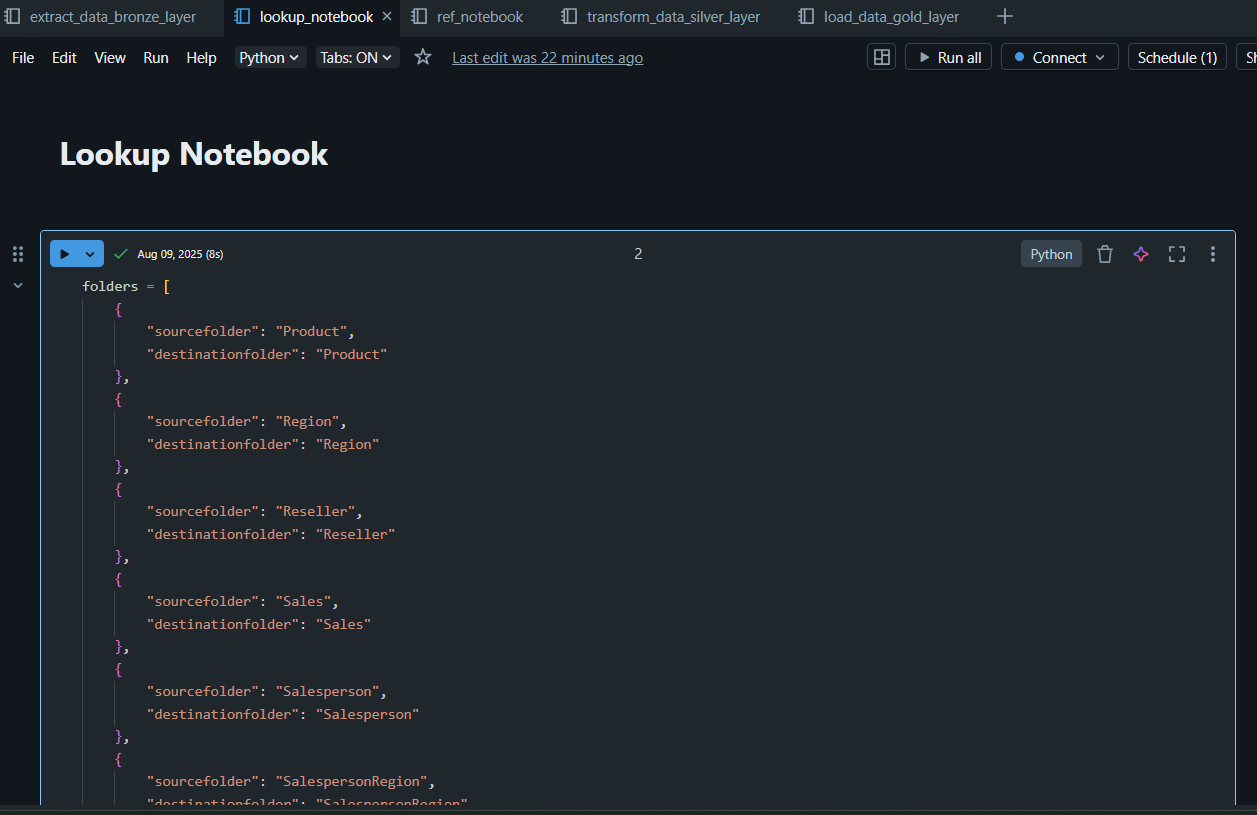
### Azure Databricks External Location



# ETL Pipeline in Azure Databricks

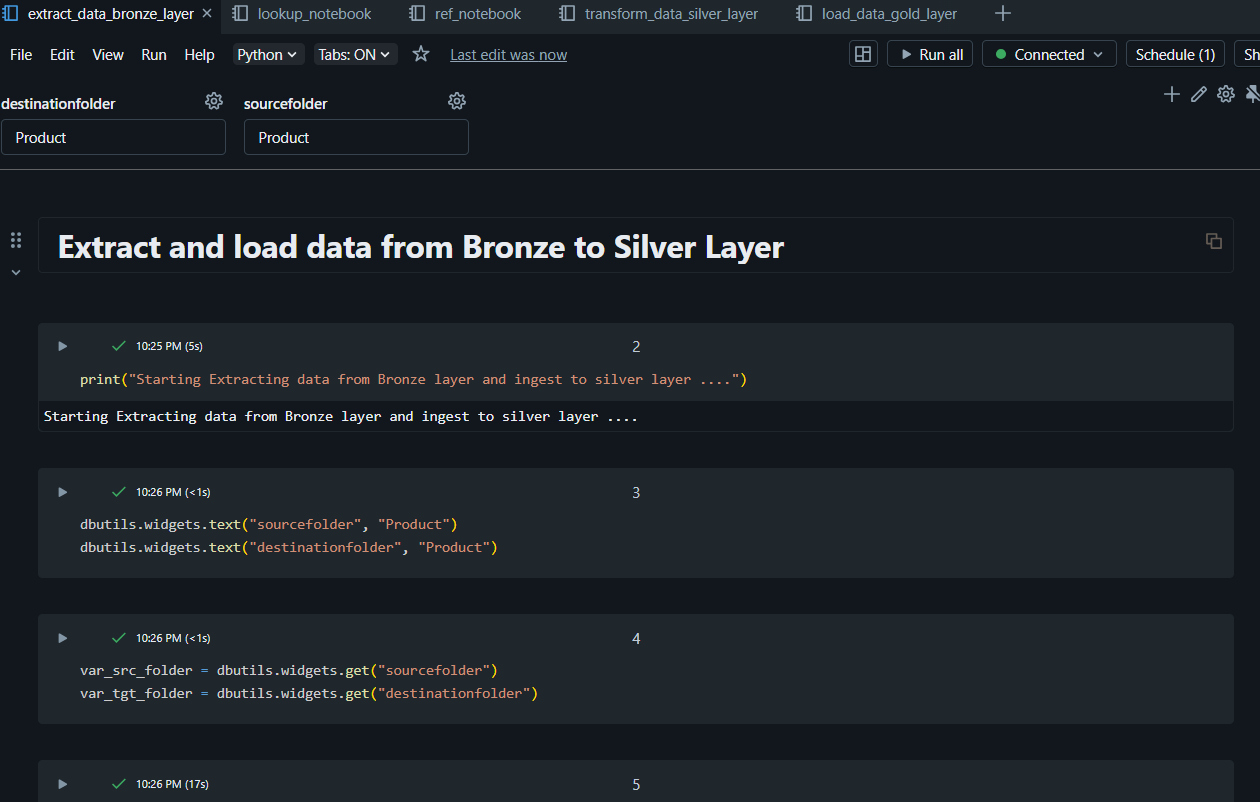
## Lookup Notebook

The Lookup Notebook is used to set a list of files and folders. The output is used in Data Transformation from Bronze to Silver Layer.



## Extract Data from Bronze Layer

Extract and Load data from Bronze layer to Silver layer:



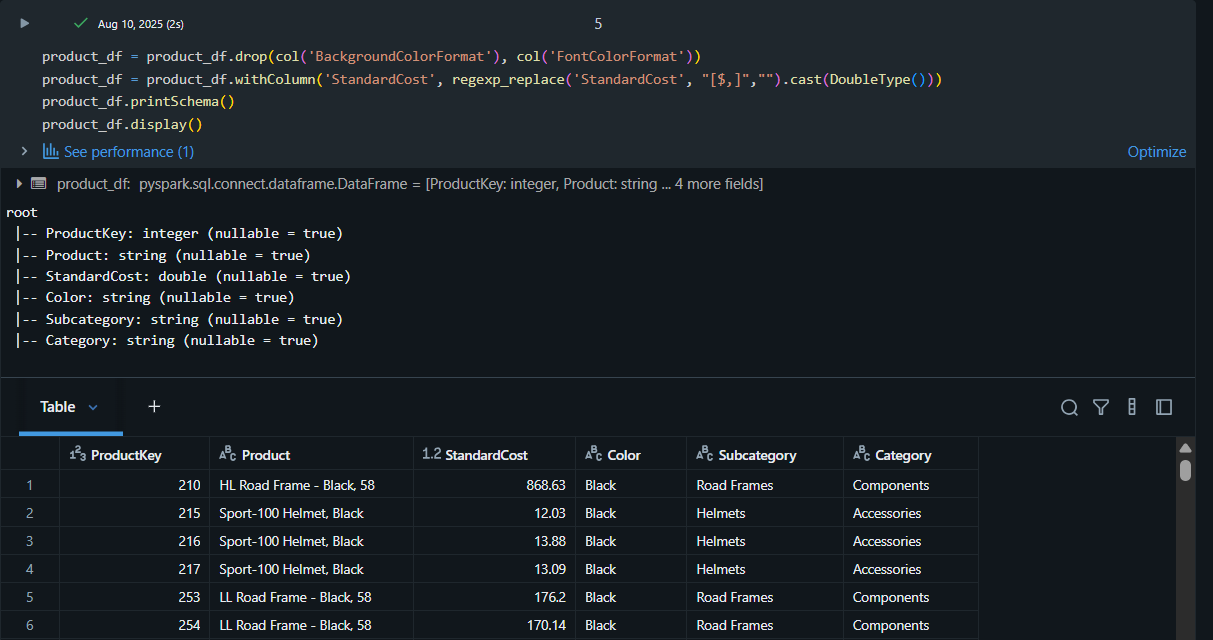
Save to Silver layer as Delta format and save in Delta Table:



## Data Transformation in Silver Layer

### Product Data:





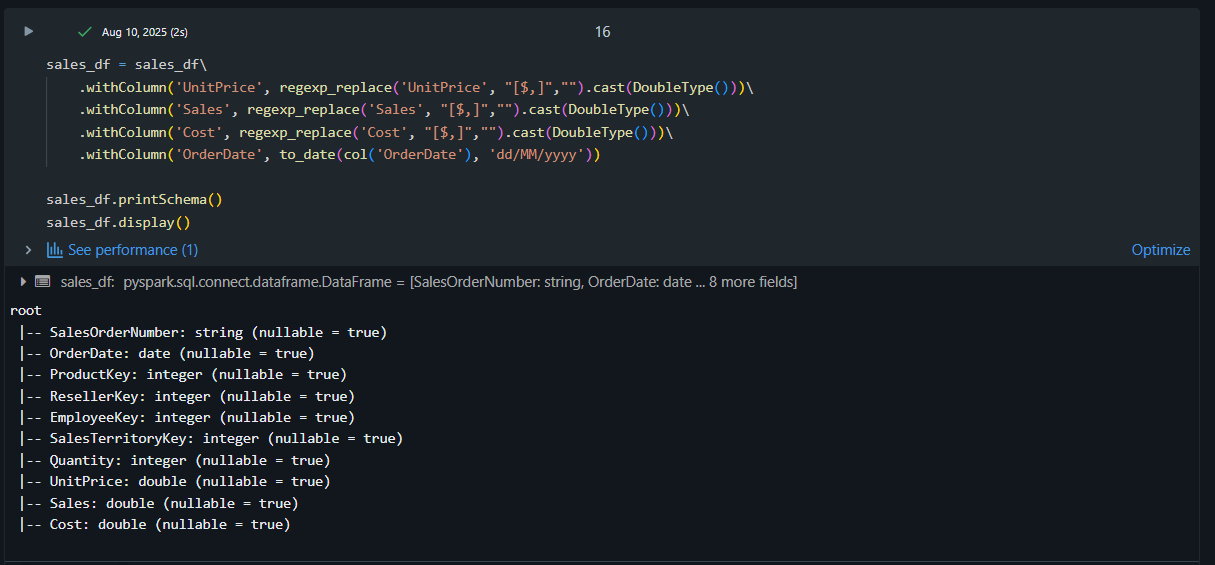
### Region Data:



### Reseller Data:

Sales Data:

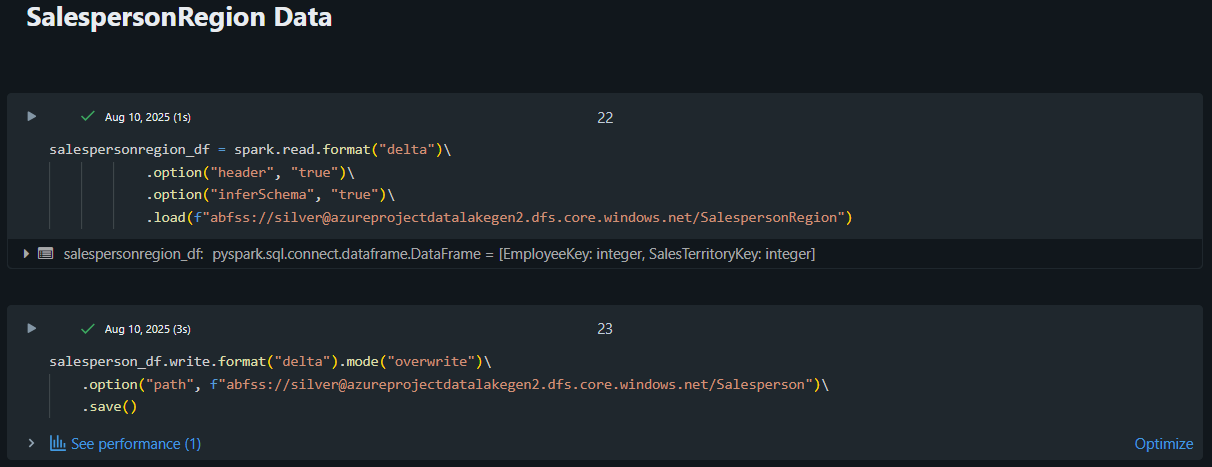




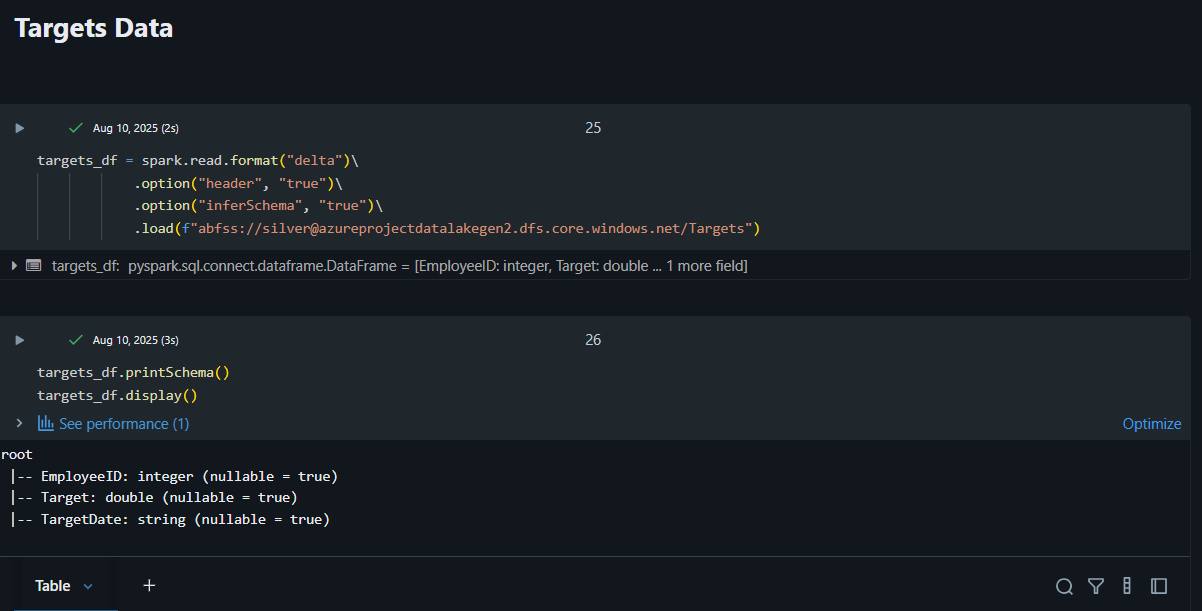
### Salesperson Data:

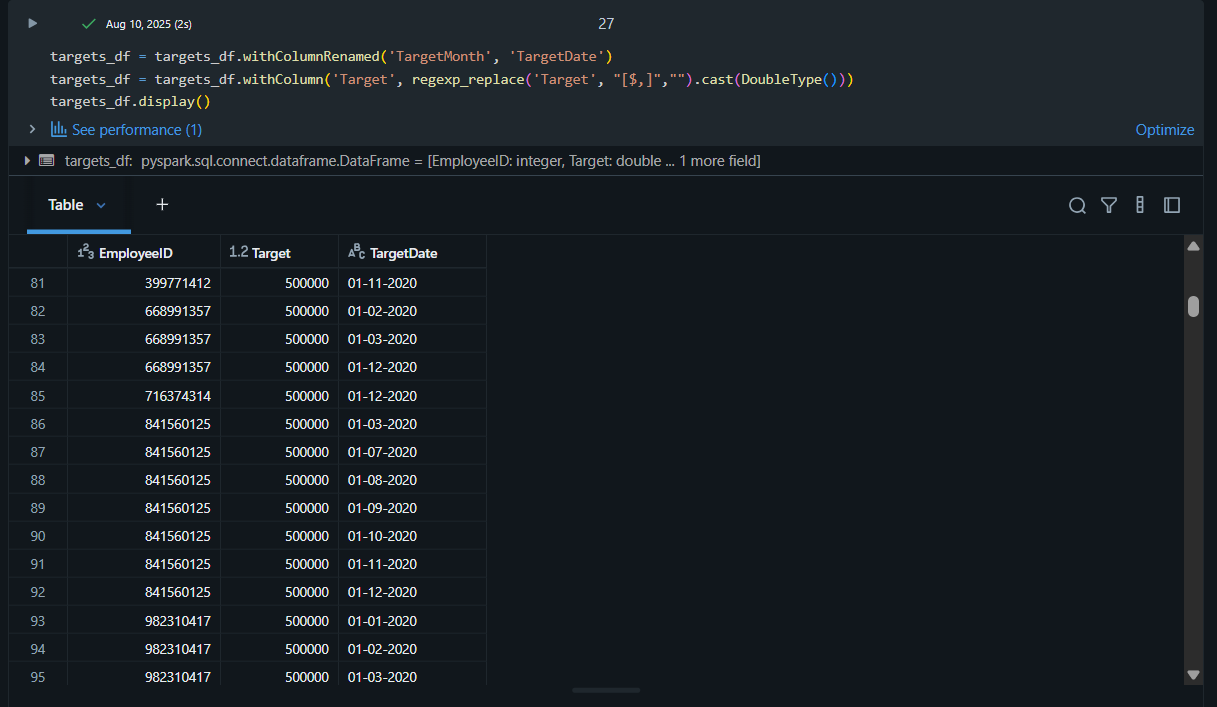


### Salesperson Region Data:



### Target Data:

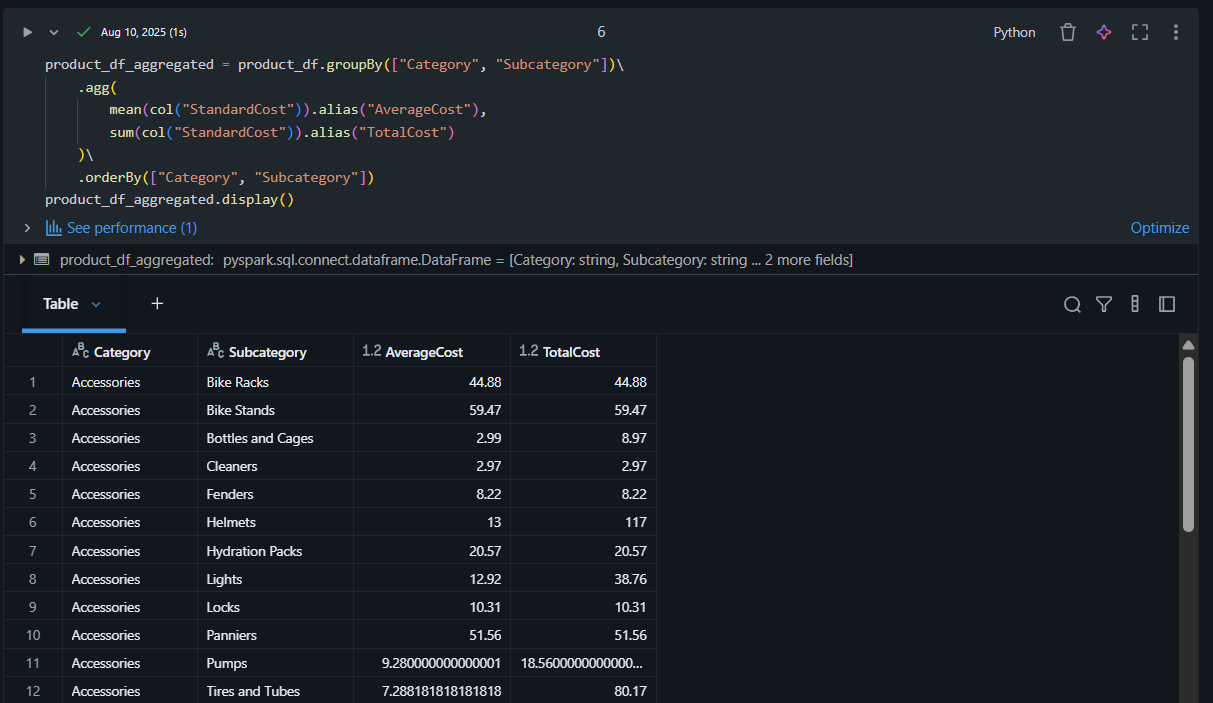




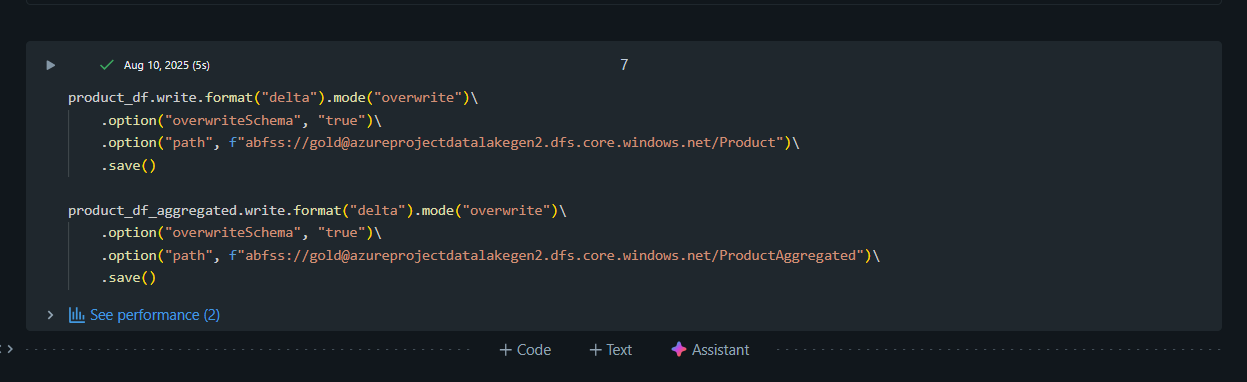
## Data Aggregation and Serving in Gold Layer

Average cost and Total Cost based on Category and Subcategory:

1. Group by Category and subcategory
2. Use mean function for average and sum function for total
3. Order by Category and Subcategory

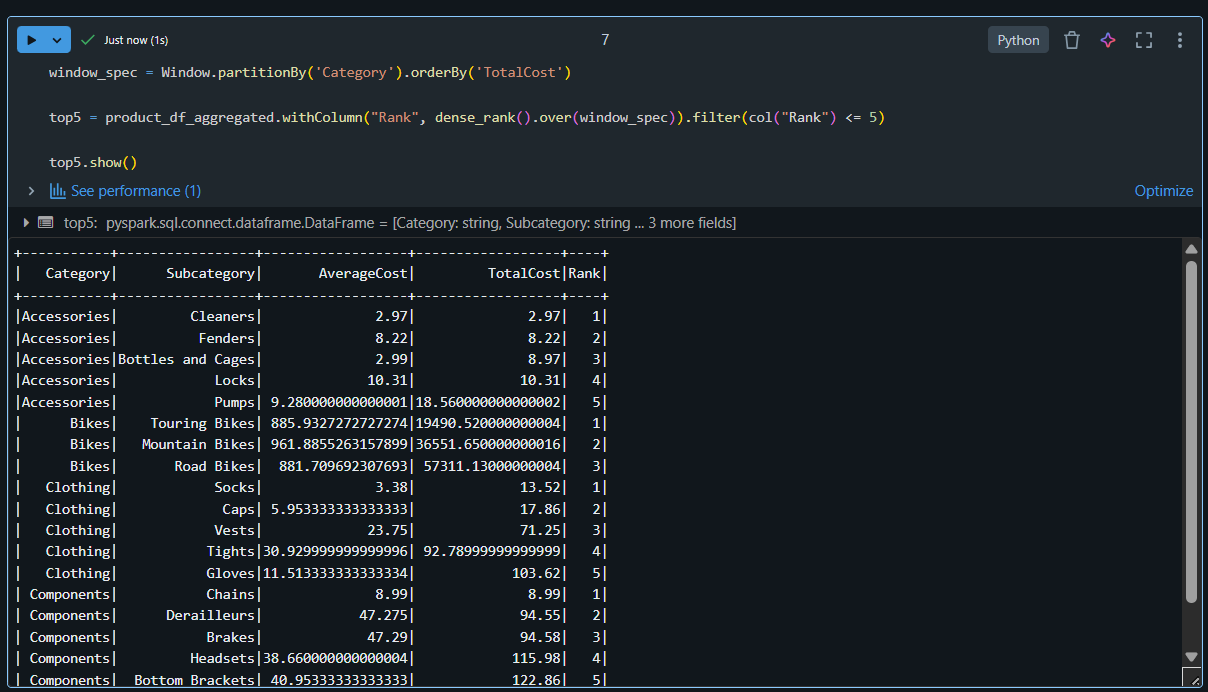


1. Save aggregated table as Delta Lake



Top 5 Products as per category

* Use window function from PySpark.
* Define window spec using partition by and order by
* Use dense\_rank() function for appropriate ranking and filter top 5 rows

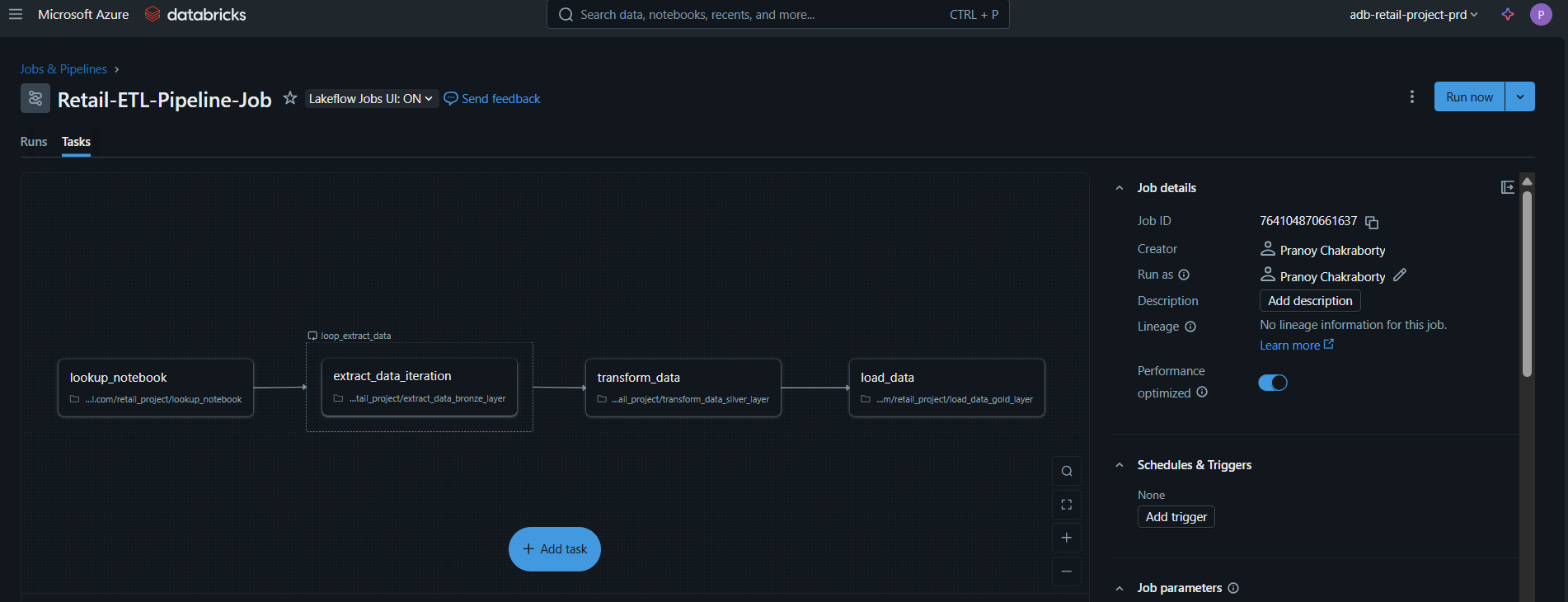


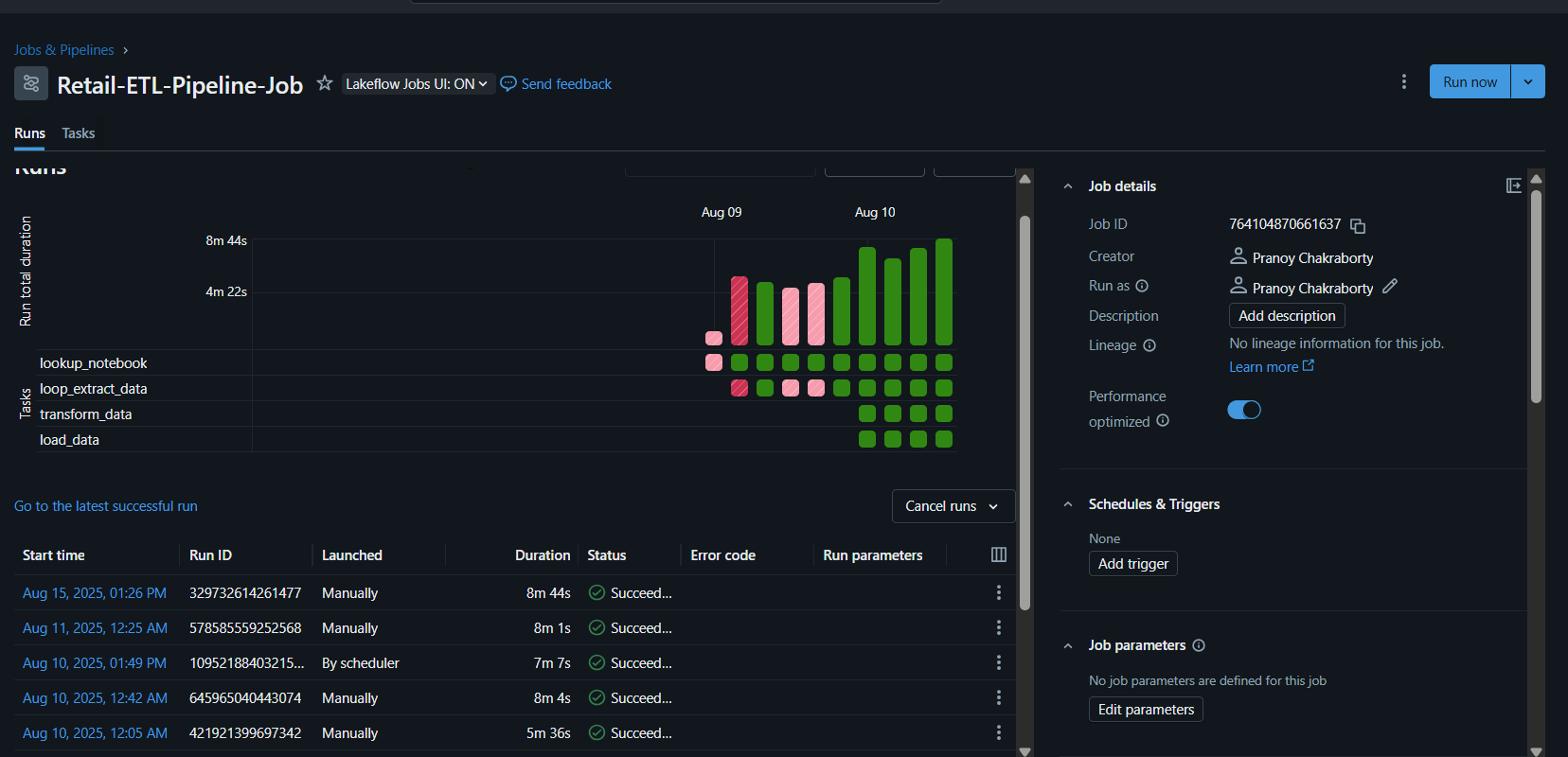
Count of cities per Country



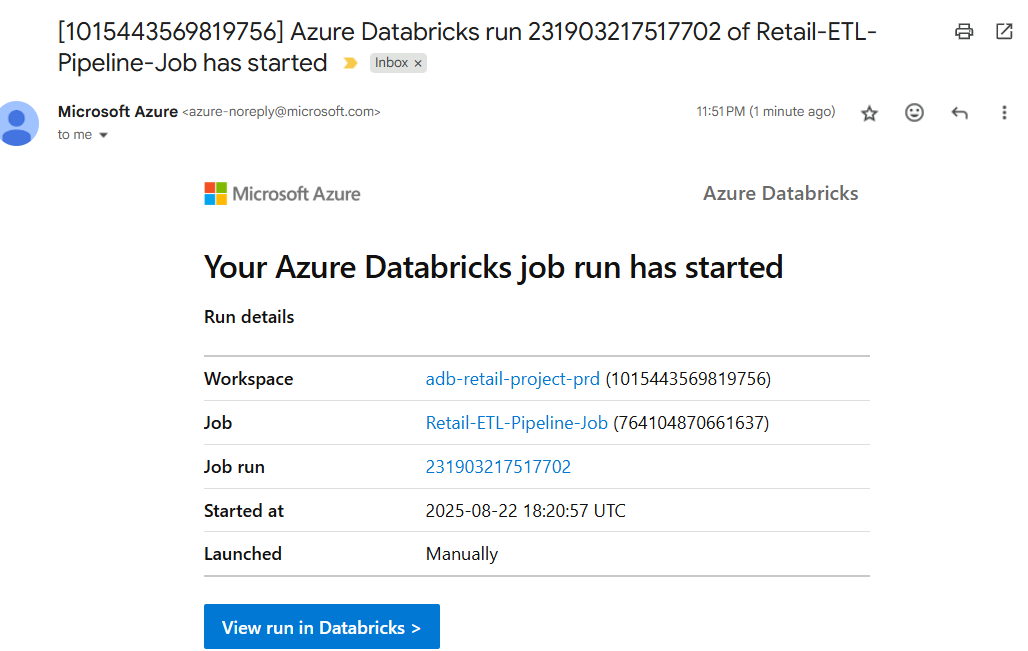
## Retail-ETL-Pipeline-Job

### Lakeflow Jobs UI

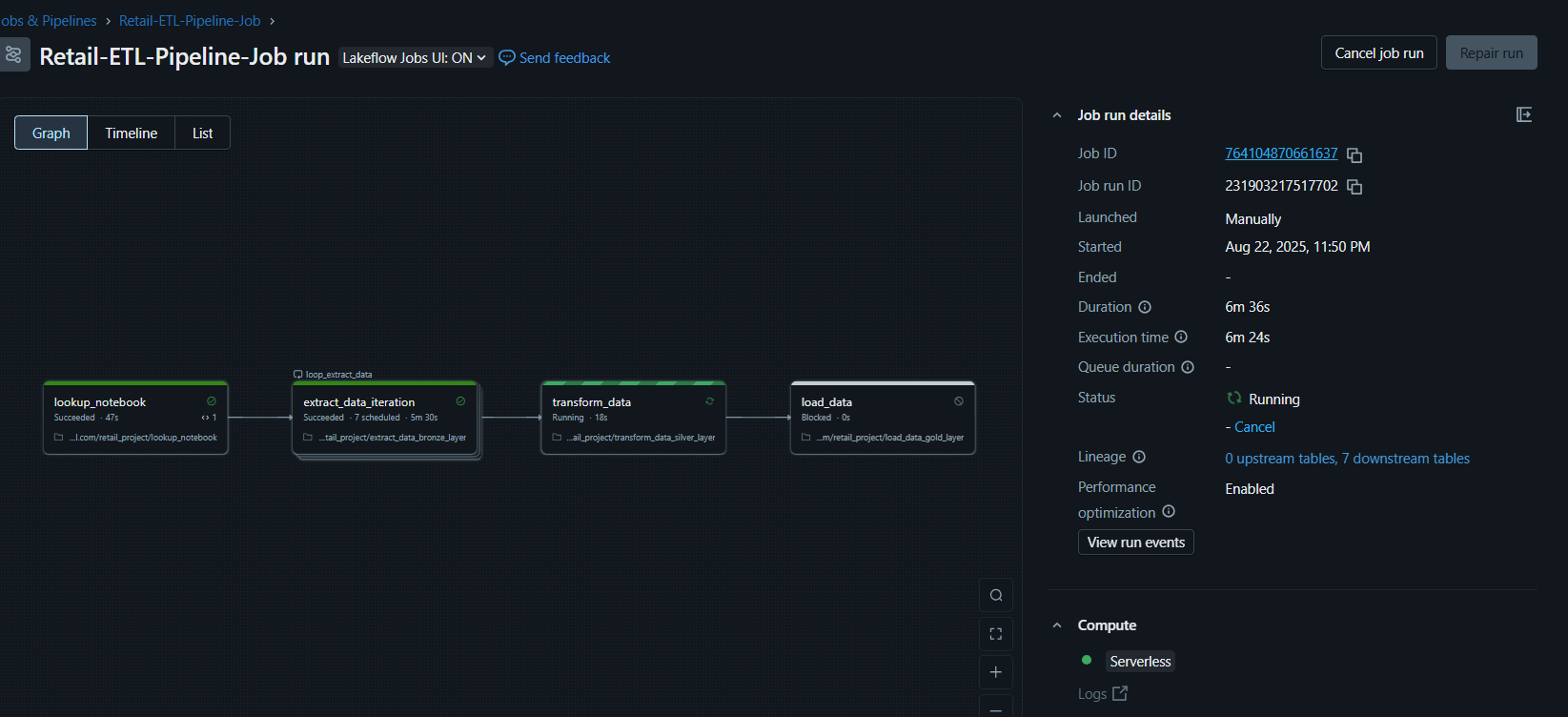




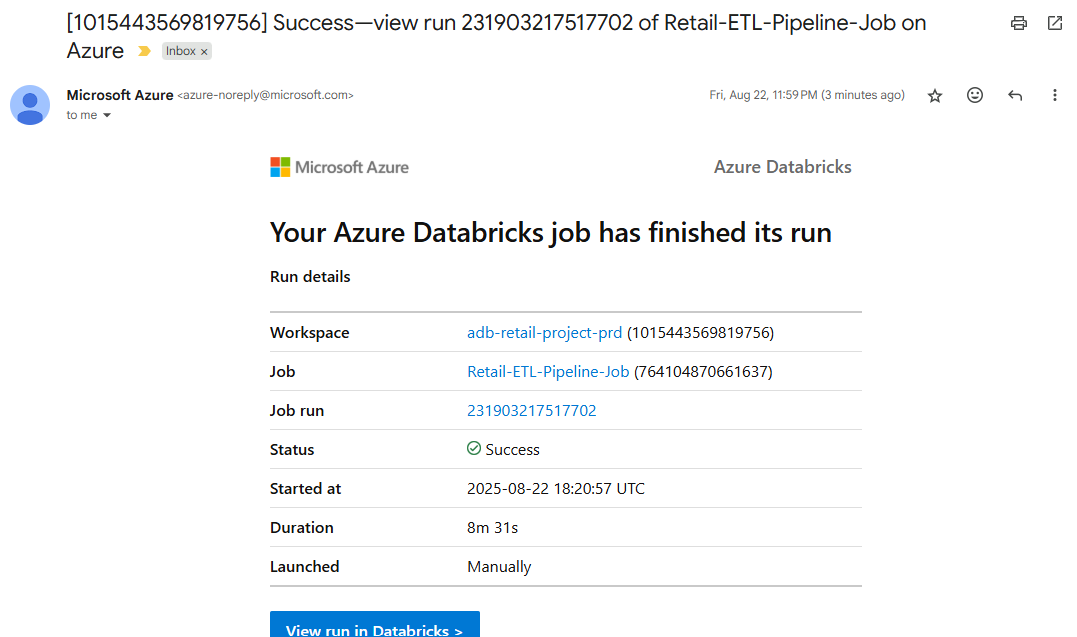
### Job Started Notification:

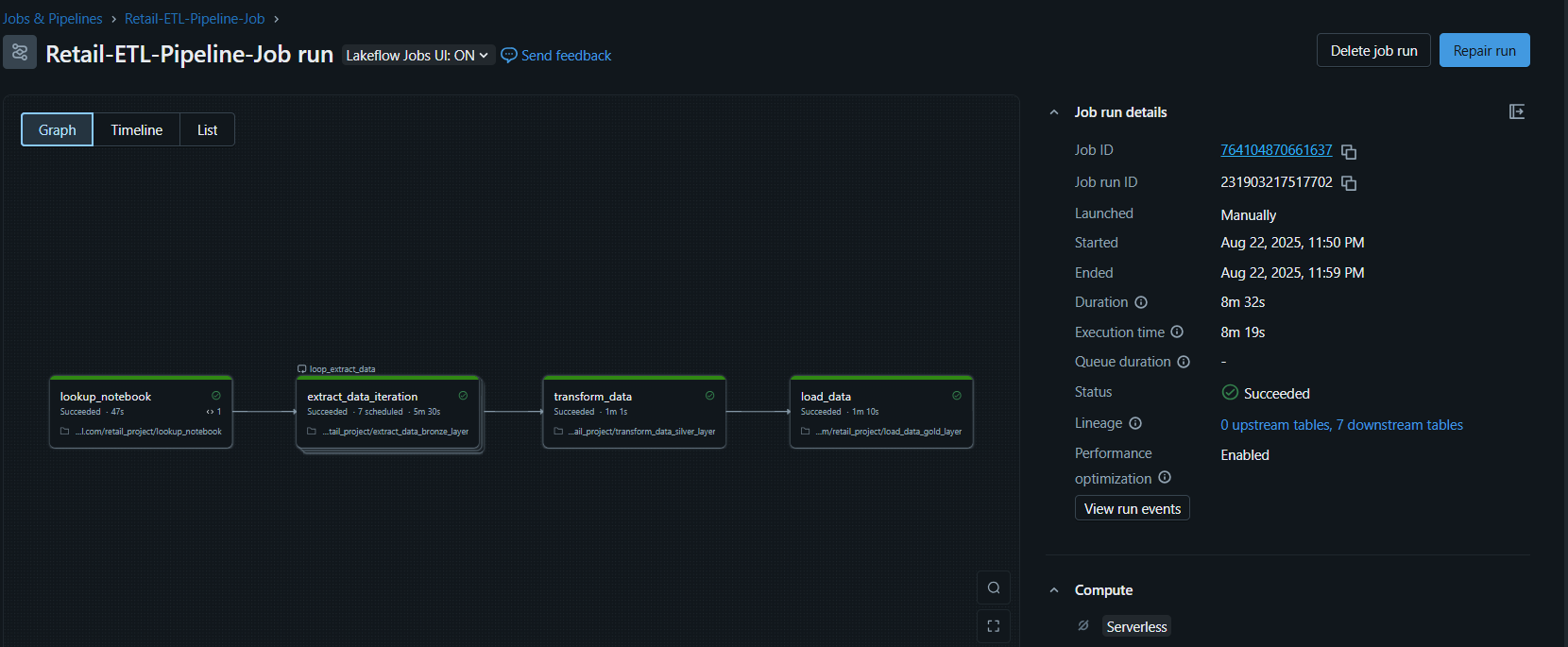


Job is running:

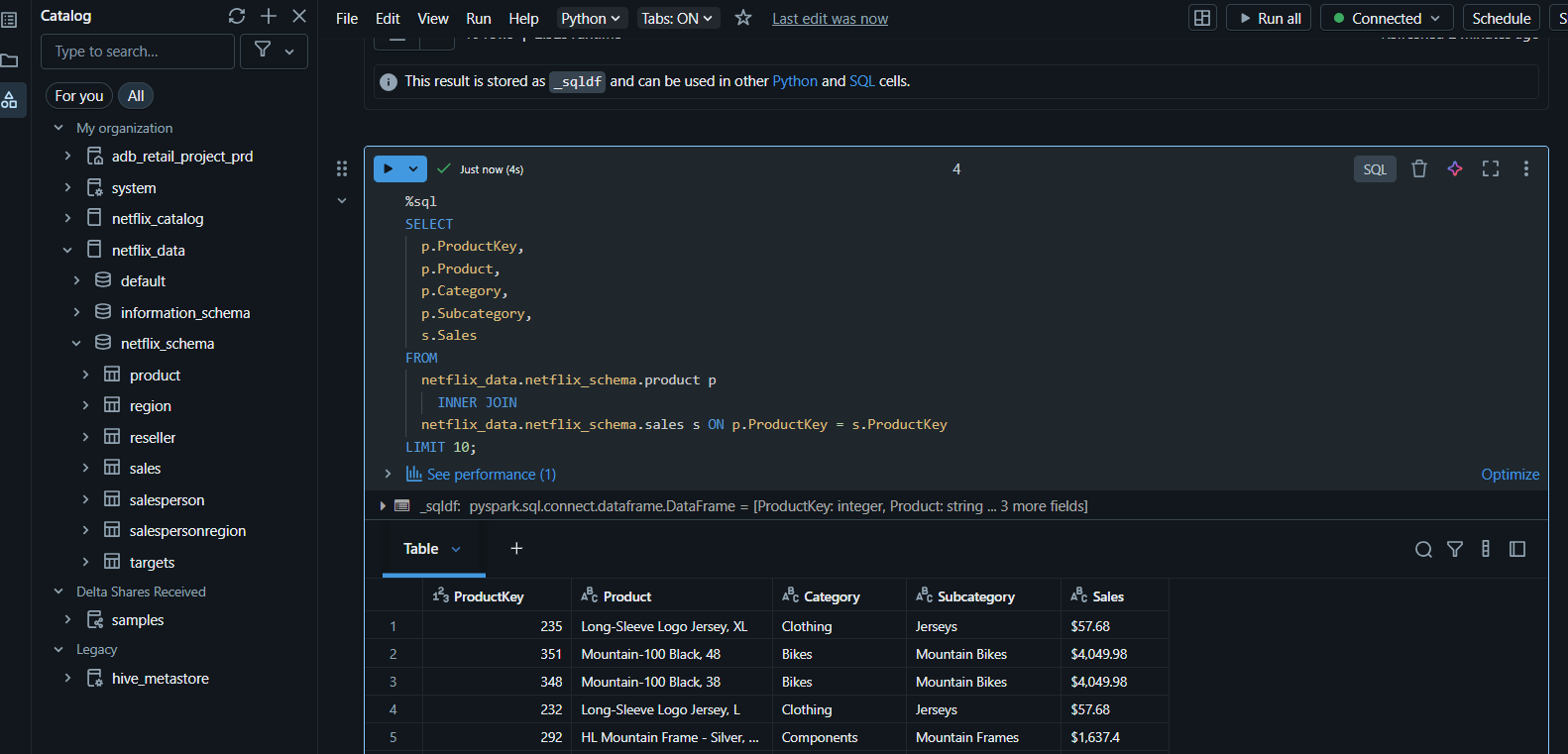


### Job Finished Notification:

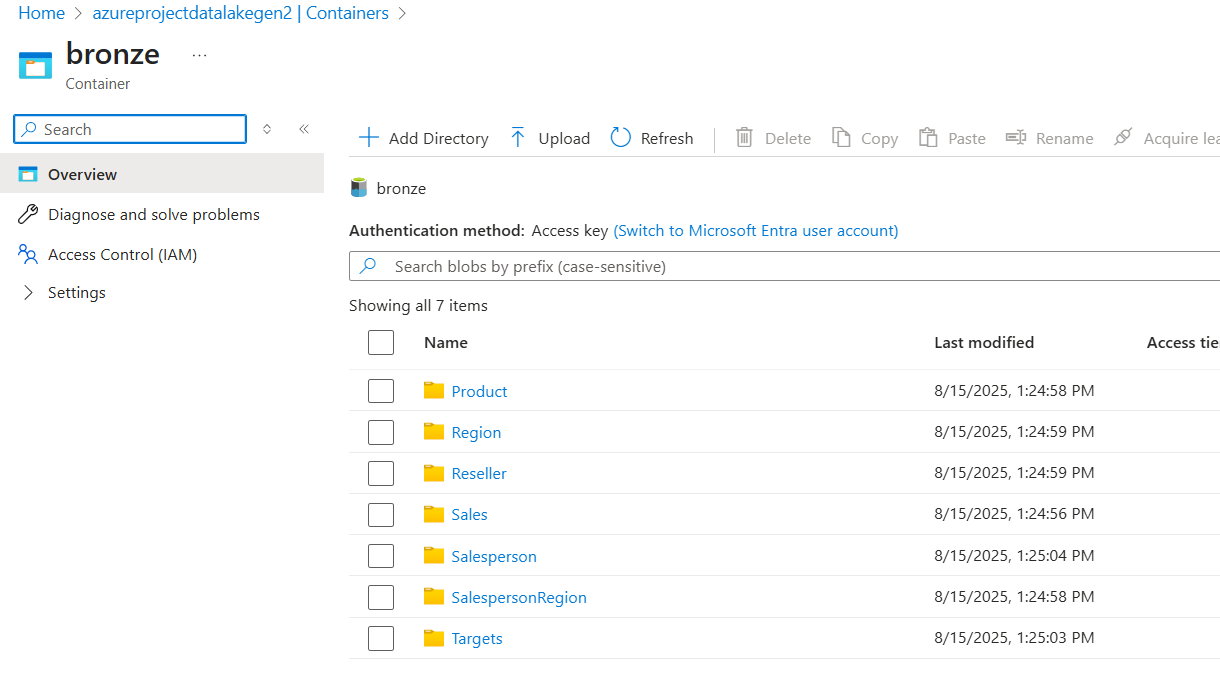




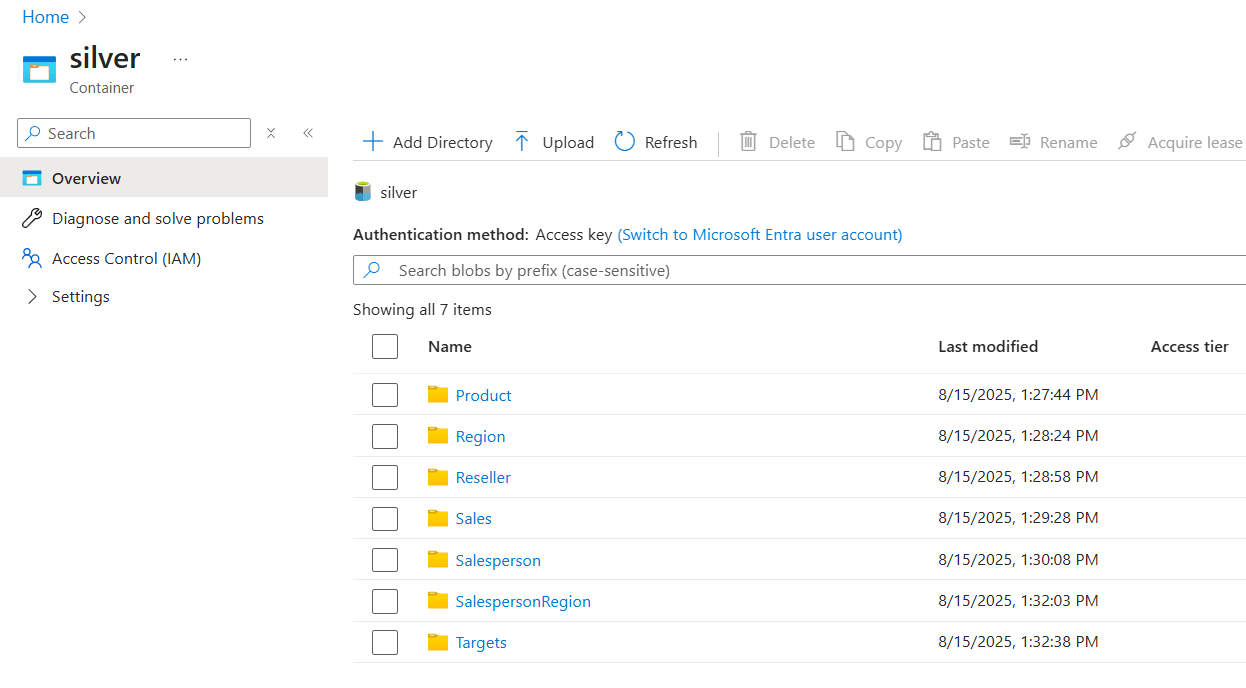
# Delta Tables in Azure Databricks



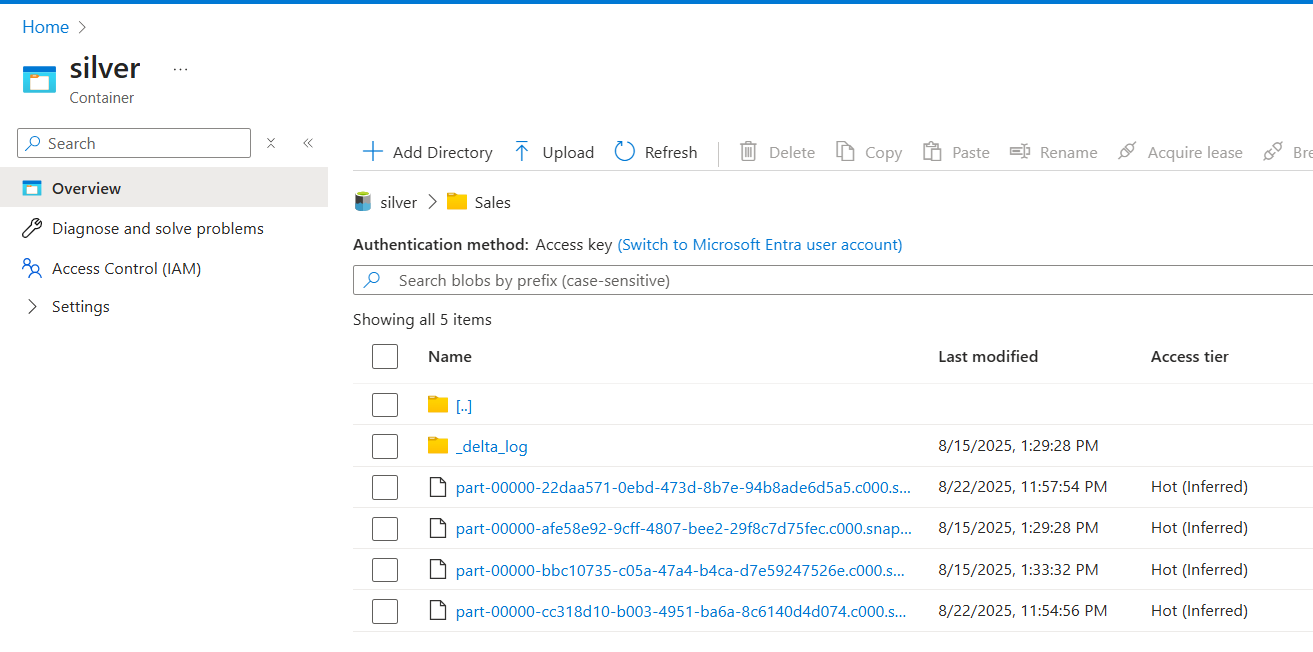
# Raw data in Bronze Layer in Azure Data Lake



# Cleaned, Enriched data stored in Silver Layer in Azure Data Lake



Data Stored in Delta Format



# Aggregated, Business Ready data Stored in Gold Layer

