NSBM Green University

Faculty of Computing

BSc (Hons) Data Science

DS403.3- Big Data Programming

Intermediate Report

Group - 2

Student ID	Student Name
24490	MRK Karunathilaka
24572	MKIM Rohana
24614	GAAS Ganegoda

Module Lecturer: Mr. Adhil Rushdy

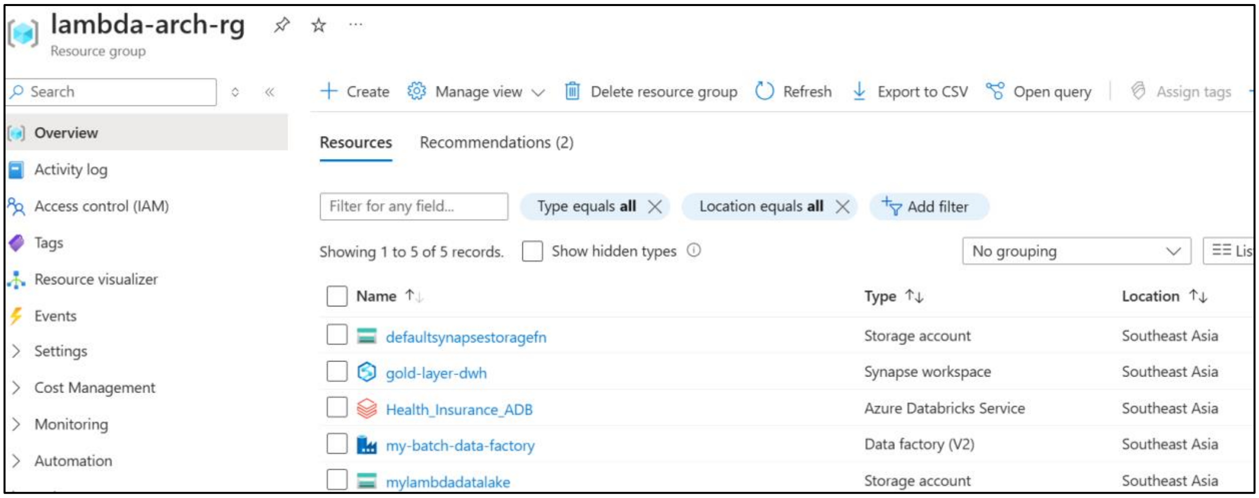
Table of Contents

1. Batch Processing Implementation Steps

Step 01: Resource Group Creation

- **Action**: Created a resource group named **lambda-arch-rg** to centralize all project resources.

- **Purpose**: Ensures organized management and cost tracking.



Type equals **all**

Location equals **all**

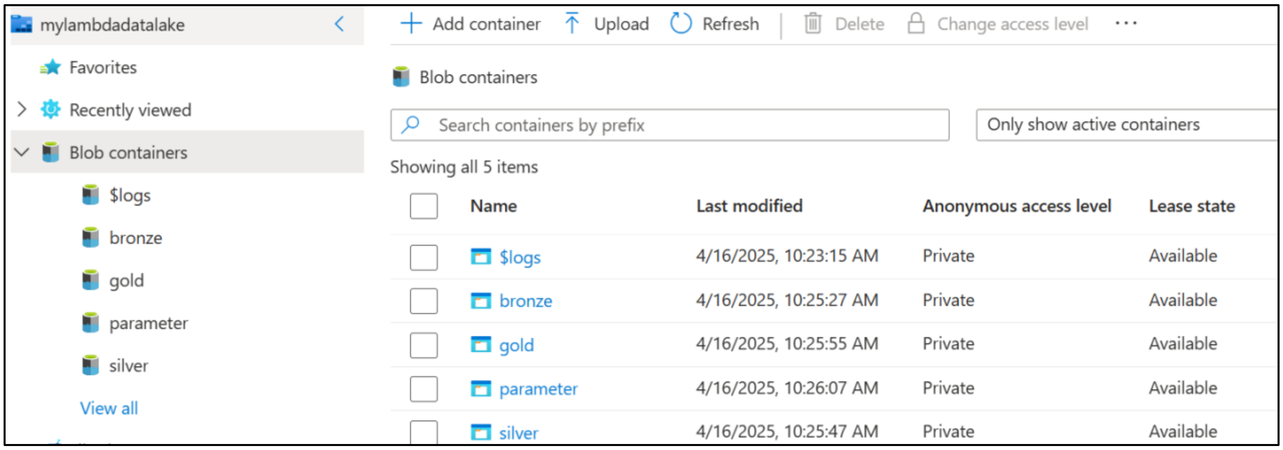
Showing 1 to 5 of 5 records.

Name	Type	Location
defaultsynapsestoragefn	Storage account	Southeast Asia
gold-layer-dwh	Synapse workspace	Southeast Asia
Health_Insurance_ADB	Azure Databricks Service	Southeast Asia
my-batch-data-factory	Data factory (V2)	Southeast Asia
mylambdadatalake	Storage account	Southeast Asia

Step 02: Data Lake Setup

- **Actions**: Created a **Storage Account** (ADLS Gen2) with four containers,
  - **bronze:** Stores raw data (e.g., CSV files from Google Drive).
  - **silver:** Holds transformed/cleaned data (Parquet format).
  - **gold:** Stores analysis-ready datasets (aggregated tables).
  - **parameter:** Contains JSON files for dynamic pipeline configurations.

- **Purpose**: A Data Lake is very cost-effective; it can store both structured and unstructured data due to its object Storage.

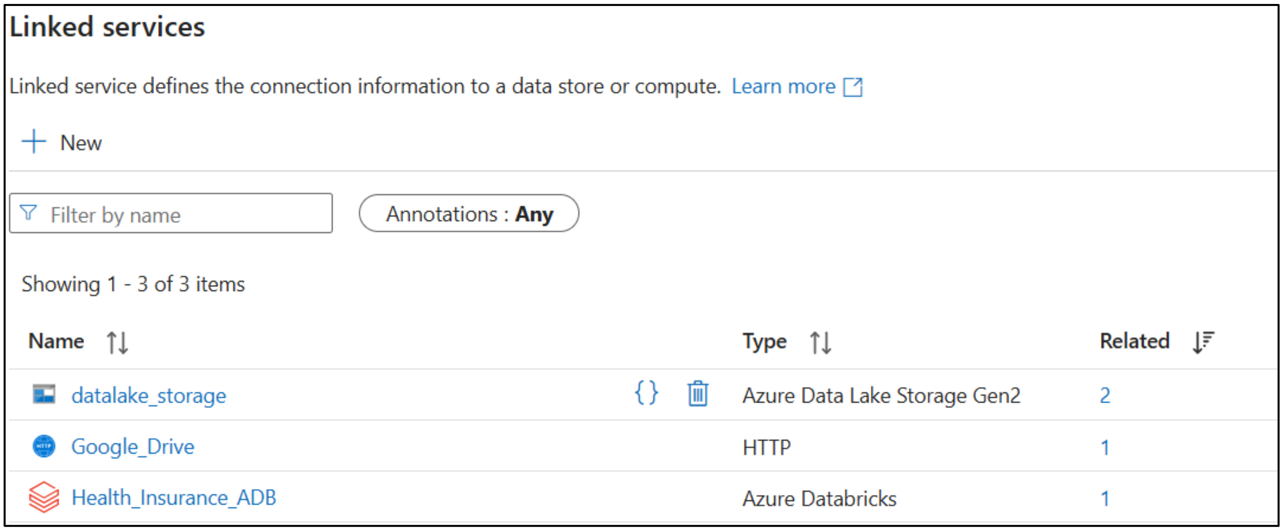


Showing all 5 items

□	Name	Last modified	Anonymous access level	Lease state
□	$logs	4/16/2025, 10:23:15 AM	Private	Available
□	bronze	4/16/2025, 10:25:27 AM	Private	Available
□	gold	4/16/2025, 10:25:55 AM	Private	Available
□	parameter	4/16/2025, 10:26:07 AM	Private	Available
□	silver	4/16/2025, 10:25:47 AM	Private	Available

Step 03: Data Ingestion with Azure Data Factory (ADF)

- **Linked Services**
  - **ADLS Gen2**: Connected to the data lake containers.
  - **Google Drive**: Enabled CSV file ingestion (fallback after Git repo failed due to file size limits).

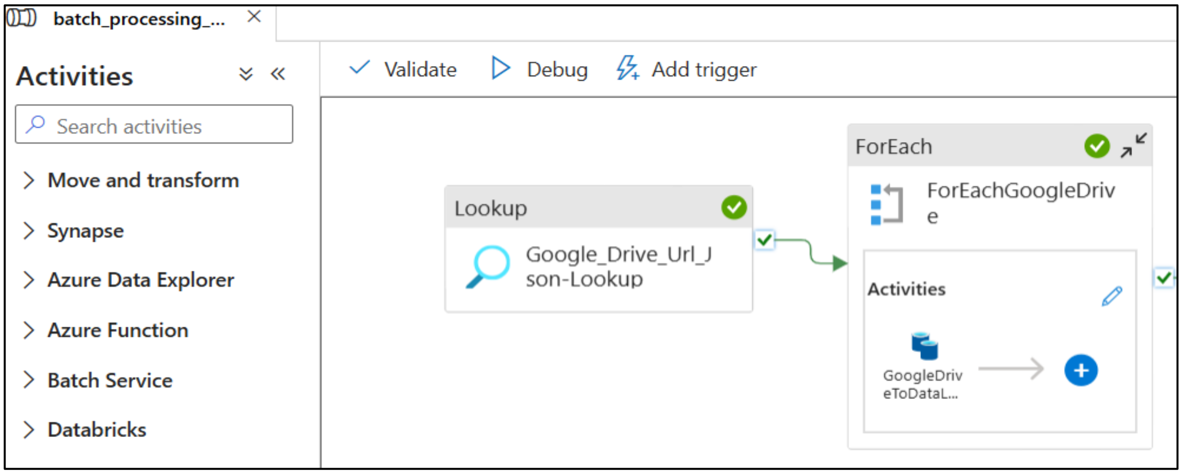

Linked services
Linked service defines the connection information to a data store or compute. Learn more
New
Filter by name
Annotations : Any

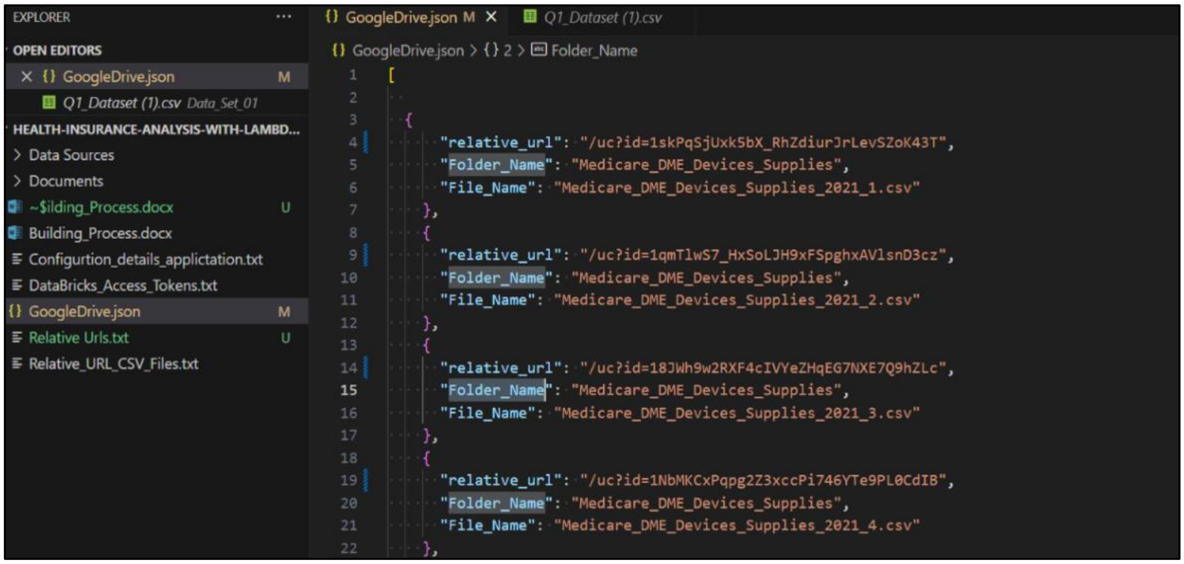
Showing 1 - 3 of 3 items

Name	Type	Related
datalake_storage	Azure Data Lake Storage Gen2	2
Google_Drive	HTTP	1
Health_Insurance_ADB	Azure Databricks	1

Challenges & Solutions

- **Google Drive 100MB Limit**
  - Split files into smaller chunks (<100MB) to avoid corruption.
- **Dynamic Pipeline**
  - Used **Lookup Activity** to fetch parameters from JSON files.
  - **ForEach Activity** + **Copy Activity** transferred files from Google Drive to bronze with,
    - Source parameter: relative_url.
    - Sink parameters: Folder_Name and File_Name.




EXPLORER
OPEN EDITORS
GoogleDrive.json M
Q1_Dataset (1).csv Data_Set_01
HEALTH-INSURANCE-ANALYSIS-WITH-LAMBD...
Data Sources
Documents
~$ilding_Process.docx U
Building_Process.docx
Configurtion_details_applictation.txt
DataBricks_Access_Tokens.txt
GoogleDrive.json M
Relative Urls.txt U
Relative_URL_CSV_Files.txt
GoogleDrive.json M
Q1_Dataset (1).csv
GoogleDrive.json > {} 2 > Folder_Name

```
[

  {
      "relative_url": "/uc?id=1skPqSjUxk5bX_RhZdiurJrLevSZoK43T",
      "Folder_Name": "Medicare_DME_Devices_Supplies",
      "File_Name": "Medicare_DME_Devices_Supplies_2021_1.csv"
    },
    {
      "relative_url": "/uc?id=1qmTlwS7_HxSoLJH9xFSpghxAVlsnD3cz",
      "Folder_Name": "Medicare_DME_Devices_Supplies",
      "File_Name": "Medicare_DME_Devices_Supplies_2021_2.csv"
    },
    {
      "relative_url": "/uc?id=18JWh9w2RXF4cIVYeZHqEG7NXE7Q9hZLc",
      "Folder_Name": "Medicare_DME_Devices_Supplies",
      "File_Name": "Medicare_DME_Devices_Supplies_2021_3.csv"
    },
    {
      "relative_url": "/uc?id=1NbMKCxPqpg2Z3xccPi746YTe9PL0CdIB",
      "Folder_Name": "Medicare_DME_Devices_Supplies",
      "File_Name": "Medicare_DME_Devices_Supplies_2021_4.csv"
    },
```

Step 04: Data Ingestion with Azure Data Factory (ADF)

- **Cluster Configuration**
  - **Single-Node Cluster** (Standard_DS3_v2, 14GB RAM) to minimize costs.
  - **Auto-termination**: Set to 10 minutes of inactivity.

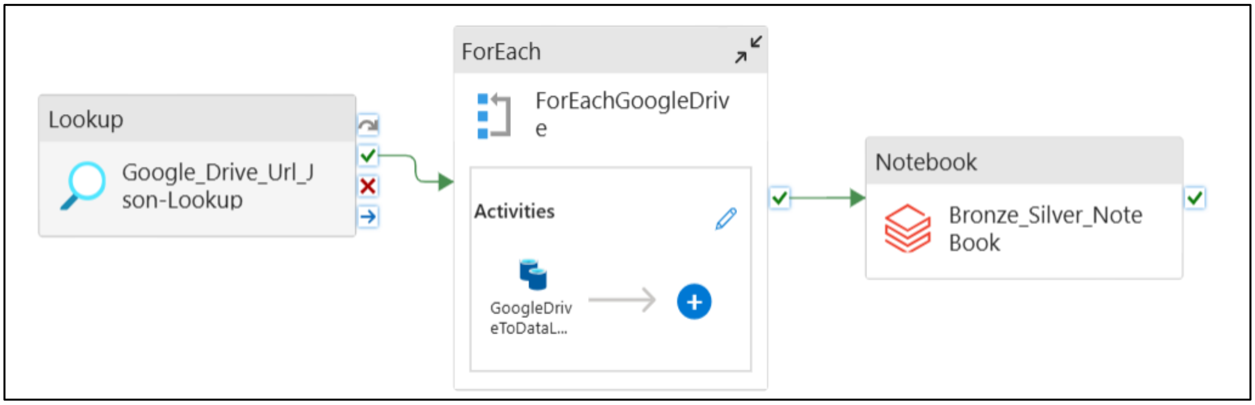
- **Data Processing**

  1. Mounted **bronze** and **silver** containers to Databricks.

  2. Loaded data into Data Frames,
     - Claims_df (claim details).
     - Drugs_df (prescription data).
     - Medicare_DME_DS_df (medical equipment records).

  3. Transformed data (cleaning) → Saved to silver (Parquet).

- **Integration**
  - Linked Databricks notebook to ADF's batch_processing_pipeline via a Databricks-linked service.



Step 05: Data Warehousing with Azure Synapse Analytics

- **Why Synapse?** Unified platform for,
  - **Data Factory (ADF)**: Pipeline orchestration (redundant with standalone ADF but retained for learning).
  - **Data Warehouse (DWH)**: Serverless SQL Pool chosen over Dedicated SQL Pool for:
    - **Cost Efficiency**: Pay-per-query (~$5/TB scanned) vs. fixed hourly costs.
    - **Data Virtualization**: Uses OPENROWSET() to query ADLS directly (no storage duplication).

2. Key Technical Decisions & Justifications in Cold Path

Component	Choice	Reason
Cluster Type	Single-Node (Databricks)	Cost savings; sufficient for batch workloads.
File Format	Parquet	Columnar storage → 80% smaller scans vs. CSV.
Synapse SQL Pool	Serverless	No infrastructure costs; scales to zero.
Data Ingestion	Google Drive + ADF	Workaround for Git’s file size limits.

3. Challenges & Solutions in Cold Path

Challenge	Solution
Google Drive file corruption (>100MB)	Split files into sub-100MB chunks.
Databricks cluster startup delays	Auto-termination + single-node configuration.
Dynamic pipeline requirements	Parameterized JSON files + Lookup Activity.