Medicare Insurance Data Ingestion Project

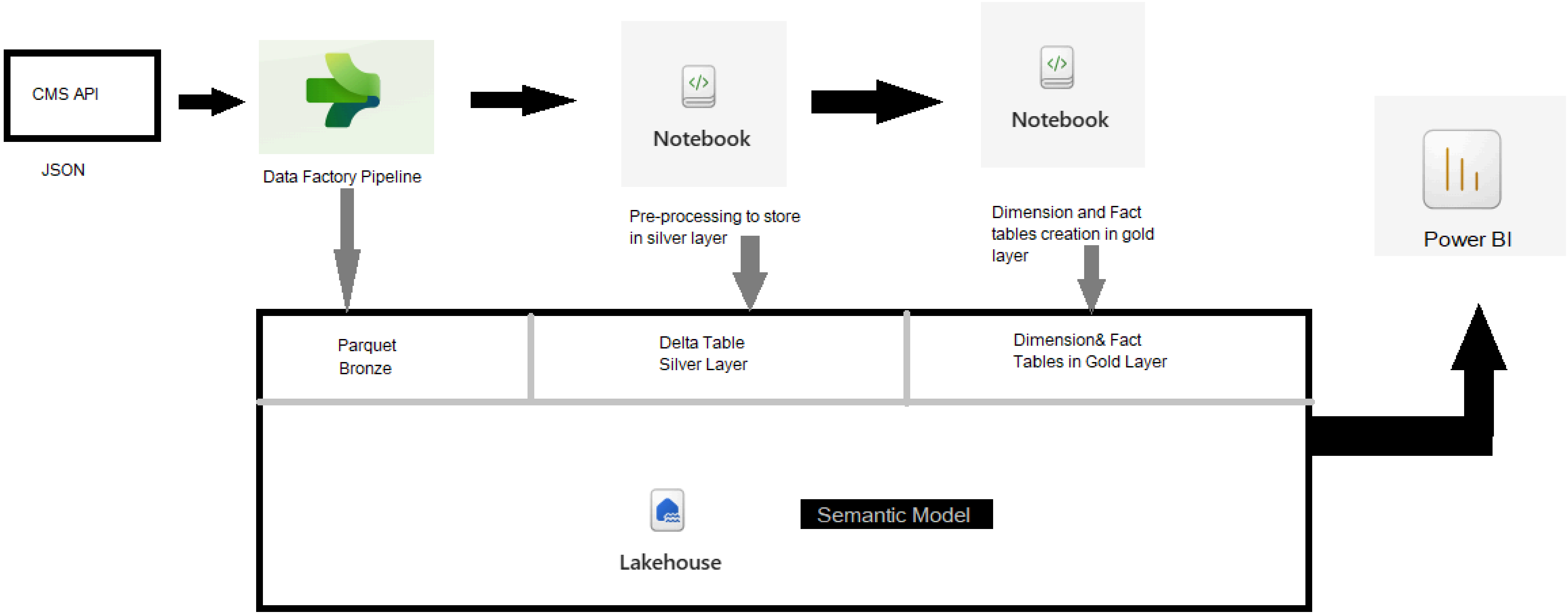
using Microsoft Fabric

MEDICARE HEALTH INSURANCE
Name/Nombre
JOHN DOE
Coverage starts/Cobertura empieza
HOSPITAL (PART A) 09-01-2017
MEDICAL (PART B) 09-01-2017

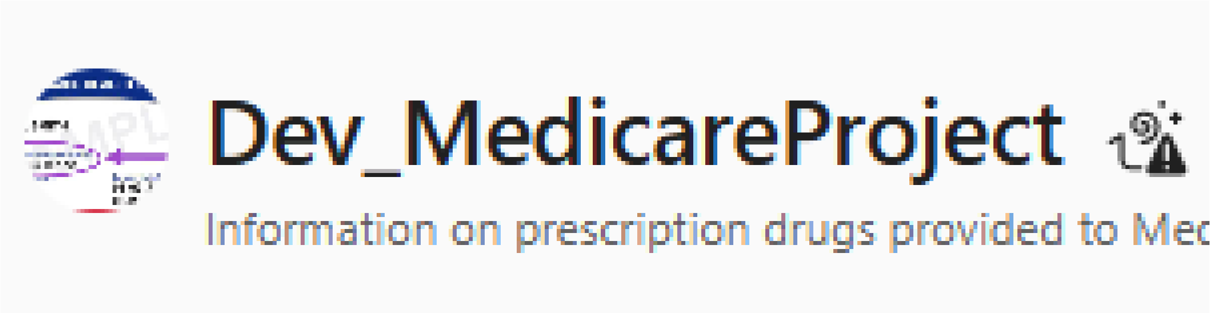
Objective

To get last 4 years Medicare’s information on prescription drugs by drug and provider incrementally into Lakehouse so that reports can be created.

Architecture



1. Create new Workspace

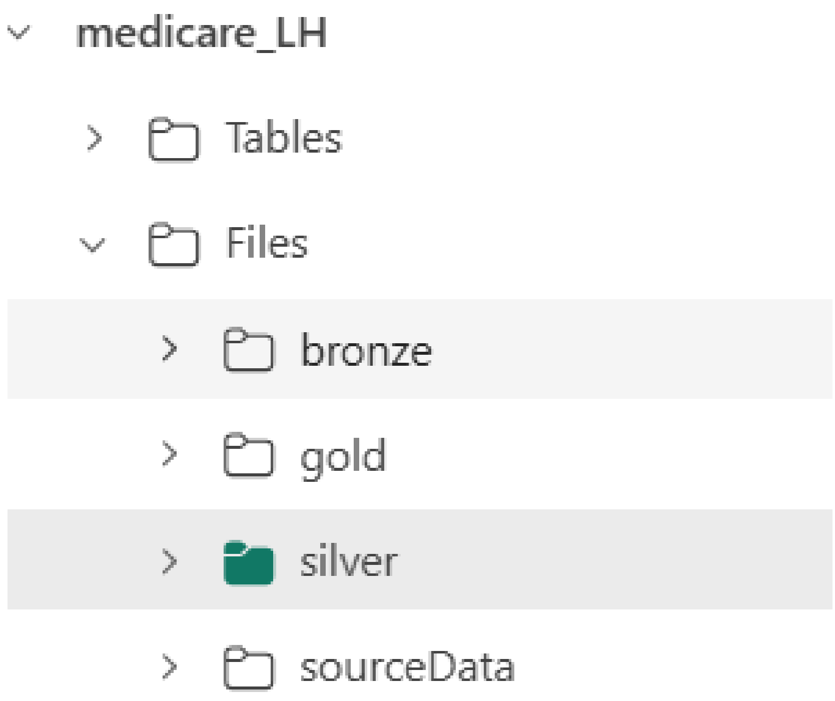


Dev_MedicareProject

Information on prescription drugs provided to Mec

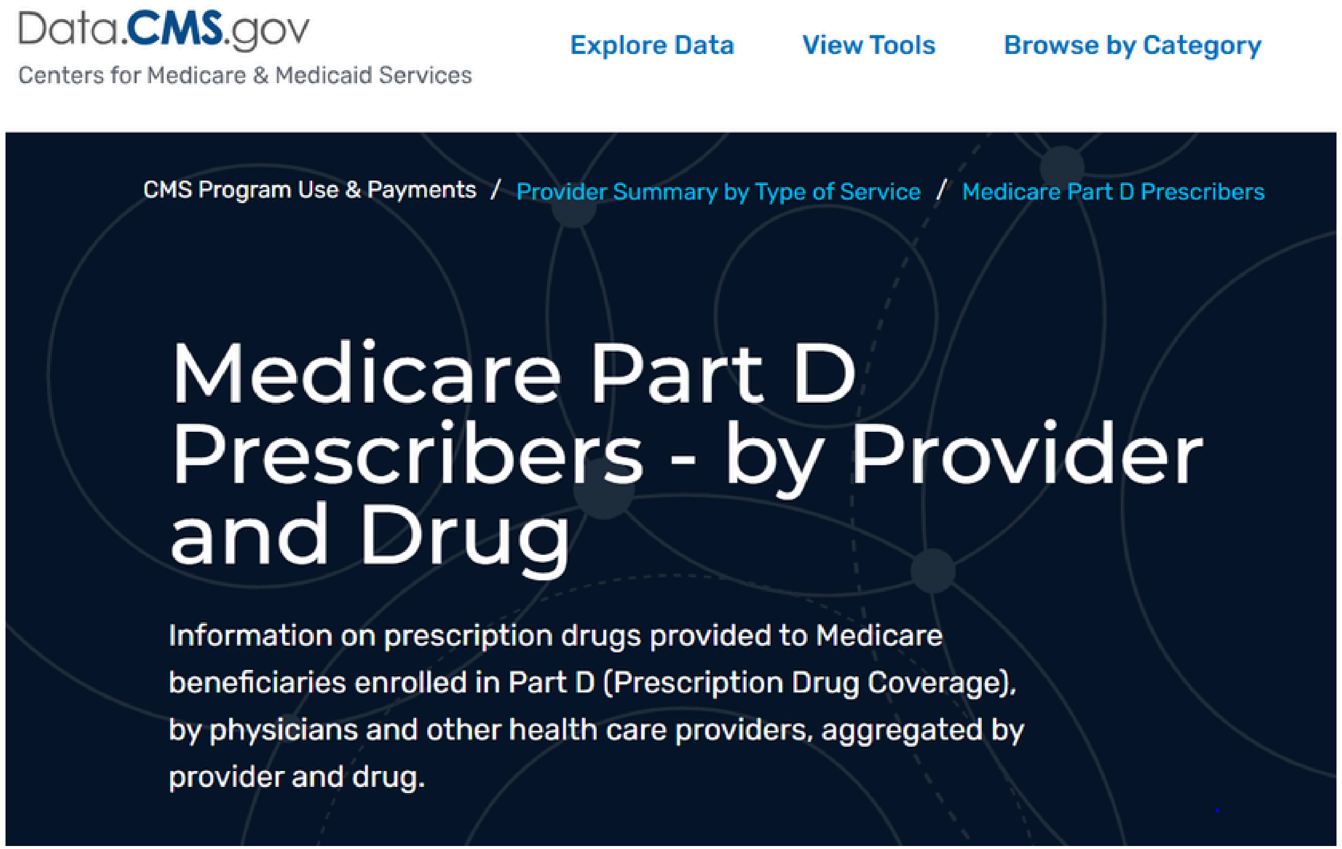
2. Add Lakehouse

Add following folders



- medicare_LH
  - Tables
  - Files
    - bronze
    - gold
    - silver
    - sourceData

3. Go to website data.cms.gov > Medicare part D prescribers - by provider and drug page



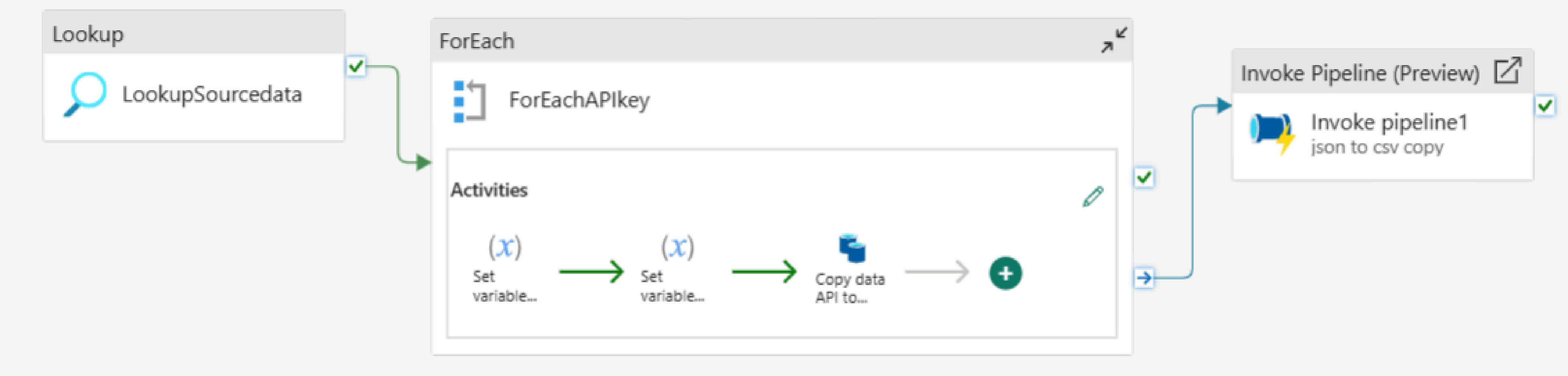
CMS Program Use & Payments / Provider Summary by Type of Service / Medicare Part D Prescribers

Medicare Part D Prescribers - by Provider and Drug

Information on prescription drugs provided to Medicare beneficiaries enrolled in Part D (Prescription Drug Coverage), by physicians and other health care providers, aggregated by provider and drug.

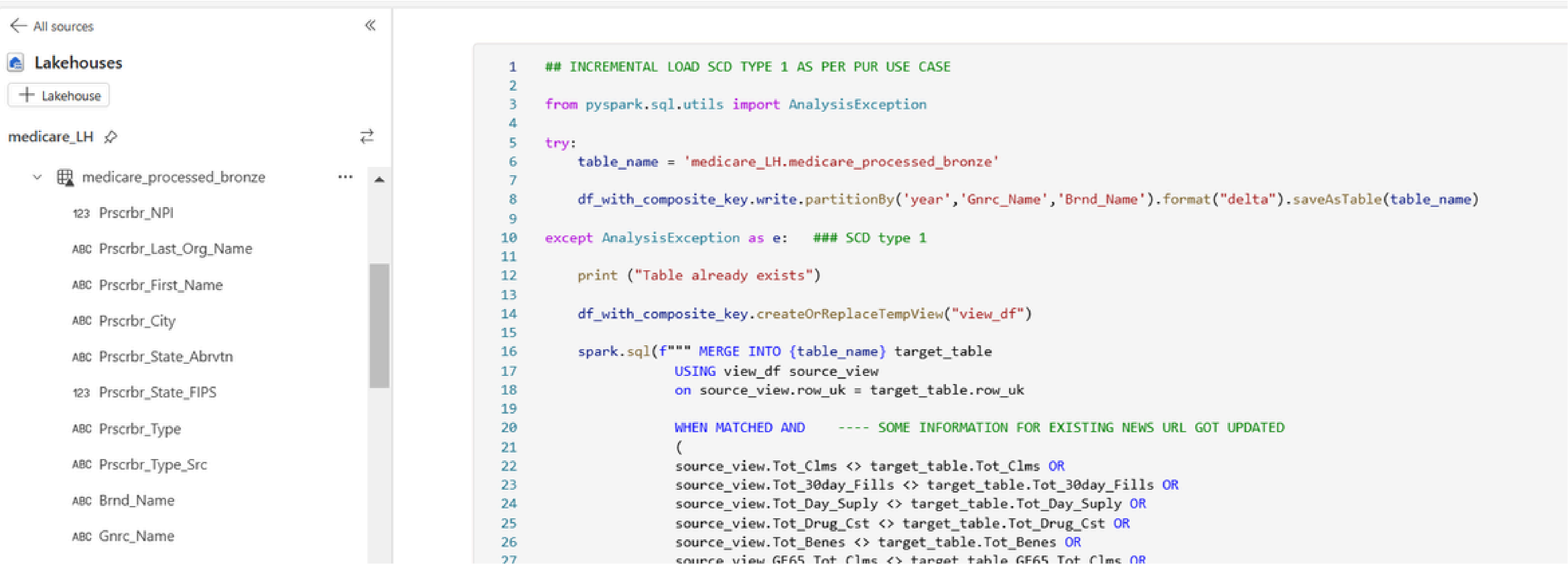
Get Access APIs year wise into a CSV file and store into sourceData directory in Lakehouse

4. Create Pipeline in the Lakehouse to copy API json format data for each file and store into bronze directory



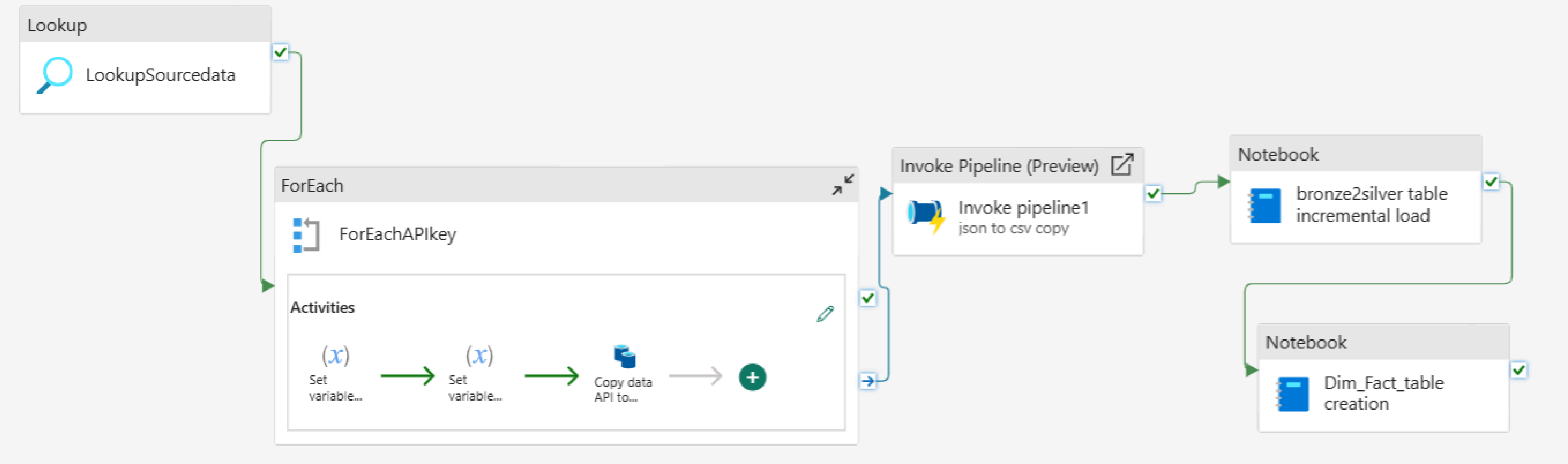
I am first copying API json format data and then converting to .csv file, but we can directly copy from API json format and leave it at that in bronze directory

5. Perform some data cleansing and then save table in delta format enabling to be incrementally upserted when notebook runs

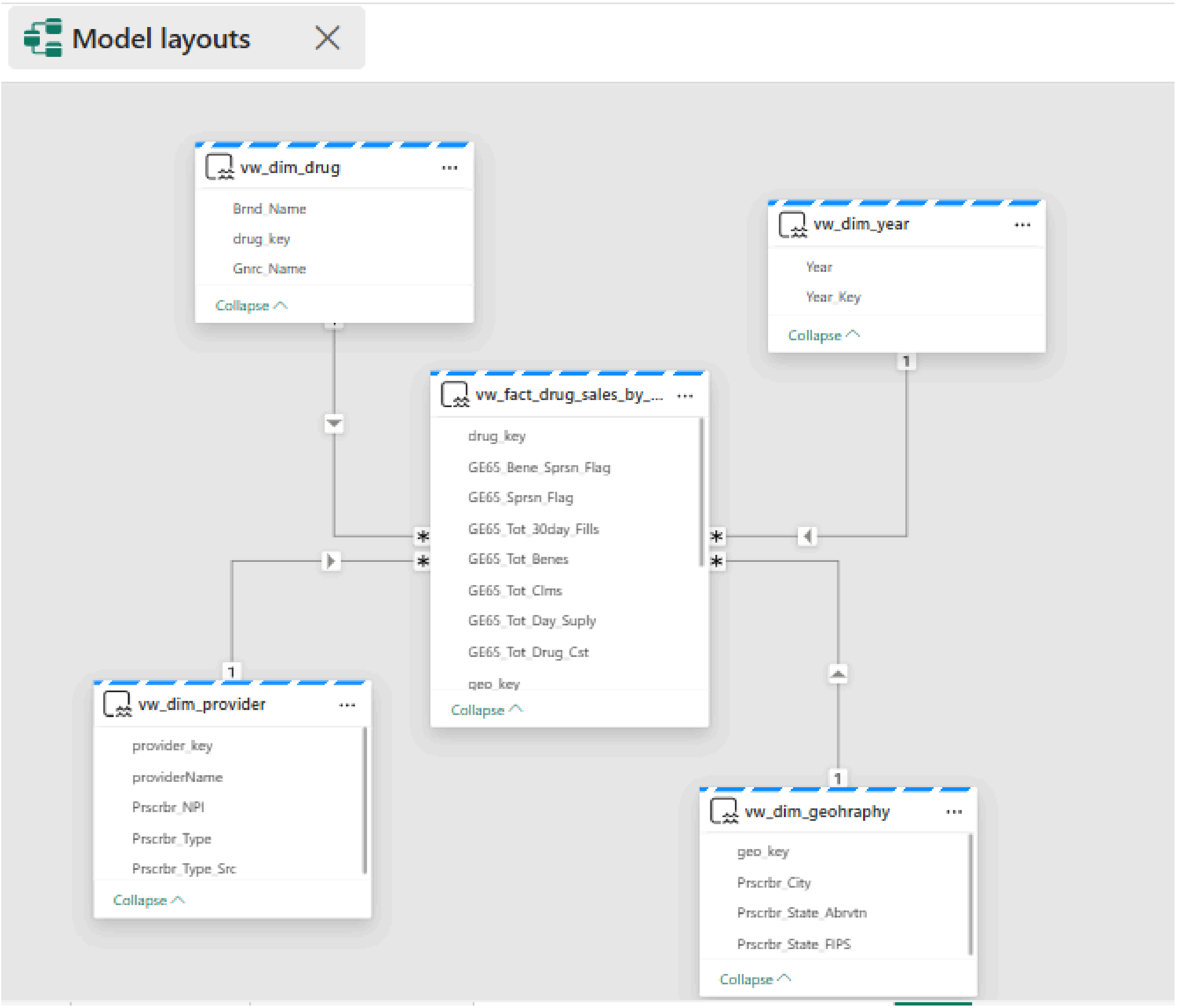


```
## INCREMENTAL LOAD SCD TYPE 1 AS PER PUR USE CASE

from pyspark.sql.utils import AnalysisException

try:
    table_name = 'medicare_LH.medicare_processed_bronze'

    df_with_composite_key.write.partitionBy('year','Gnrc_Name','Brnd_Name').format("delta").saveAsTable(table_name)

except AnalysisException as e:   ### SCD type 1

    print ("Table already exists")

    df_with_composite_key.createOrReplaceTempView("view_df")

    spark.sql(f""" MERGE INTO {table_name} target_table
                USING view_df source_view
                on source_view.row_uk = target_table.row_uk

                WHEN MATCHED AND    ---- SOME INFORMATION FOR EXISTING NEWS URL GOT UPDATED
                (
                source_view.Tot_Clms <> target_table.Tot_Clms OR
                source_view.Tot_30day_Fills <> target_table.Tot_30day_Fills OR
                source_view.Tot_Day_Suply <> target_table.Tot_Day_Suply OR
                source_view.Tot_Drug_Cst <> target_table.Tot_Drug_Cst OR
                source_view.Tot_Benes <> target_table.Tot_Benes OR
                source_view.GE65_Tot_Clms <> target_table.GE65_Tot_Clms OR
```

6. Create a new notebook to transform table into Dimension and Fact in Gold layer

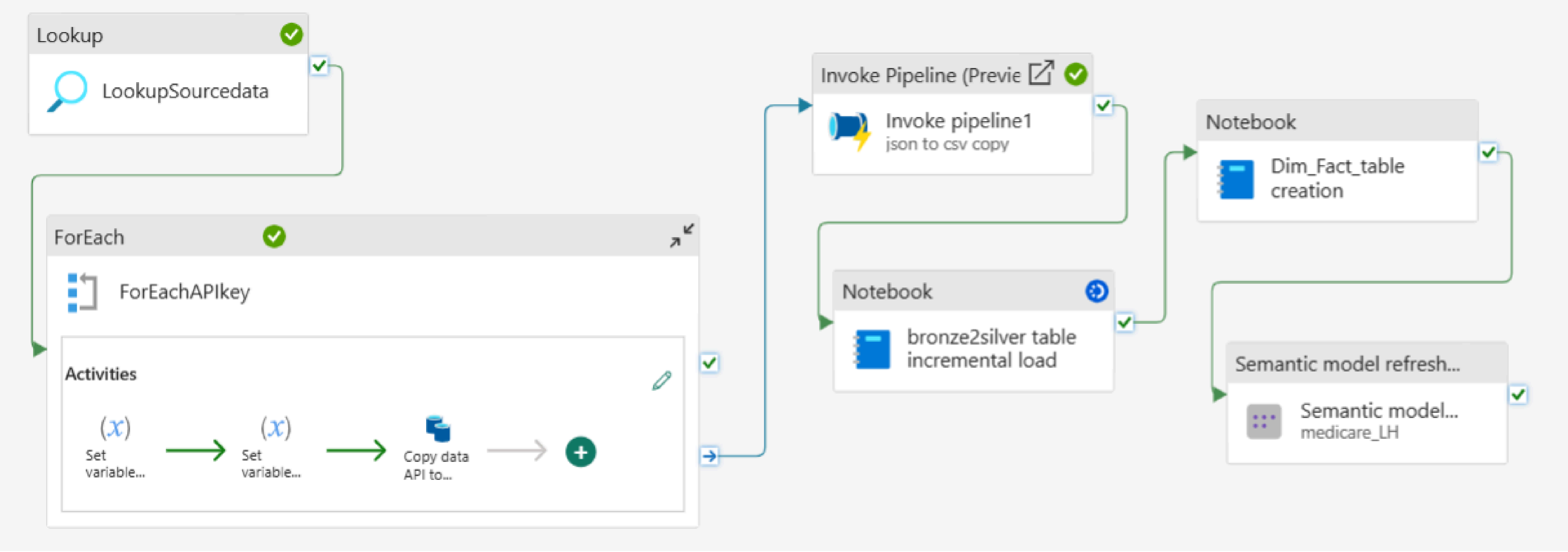


7. Adjust relationships in semantic data model layer

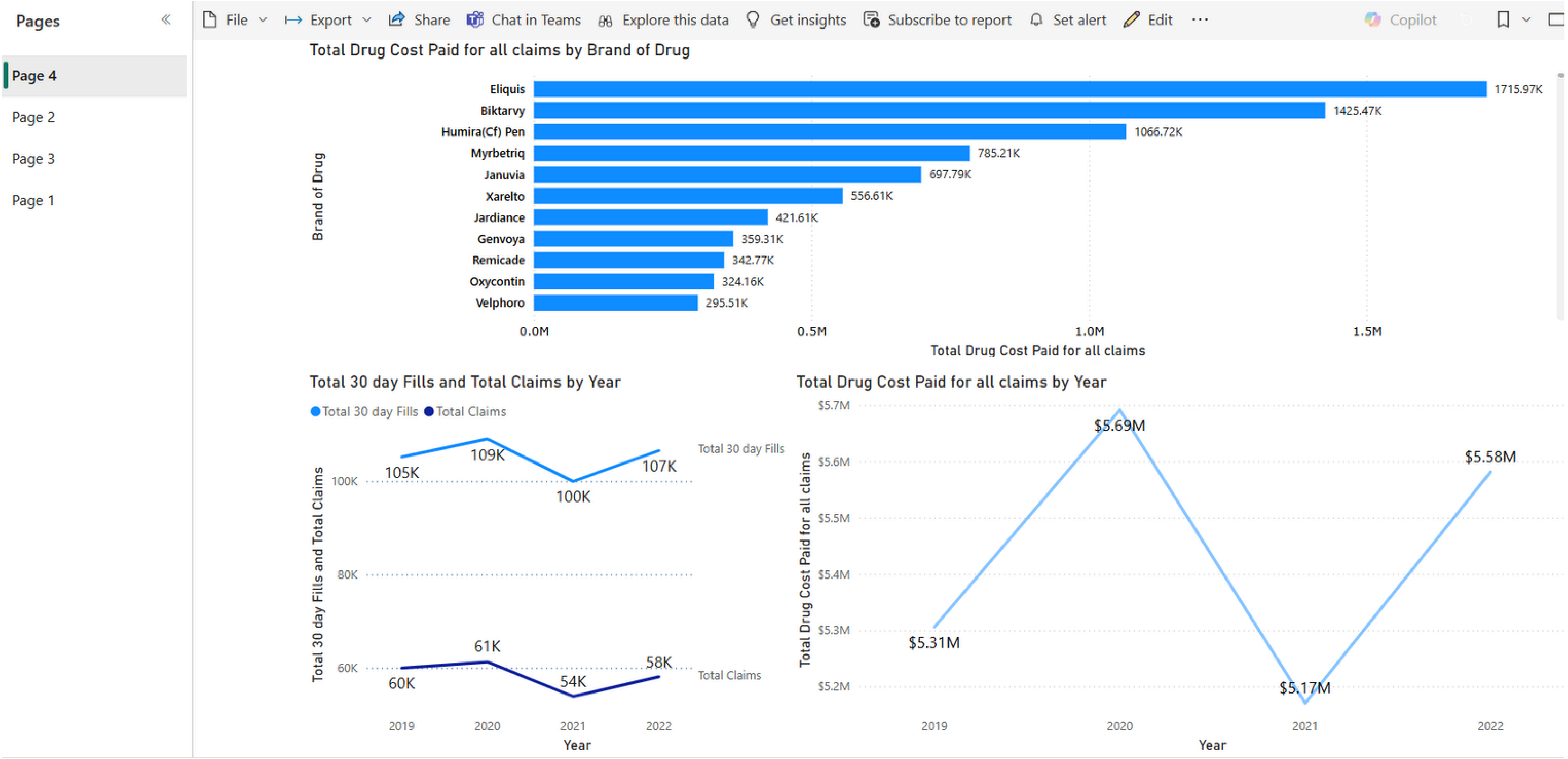


Create views on the dimension delta tables and the fact delta table using SQL analytics endpoint query

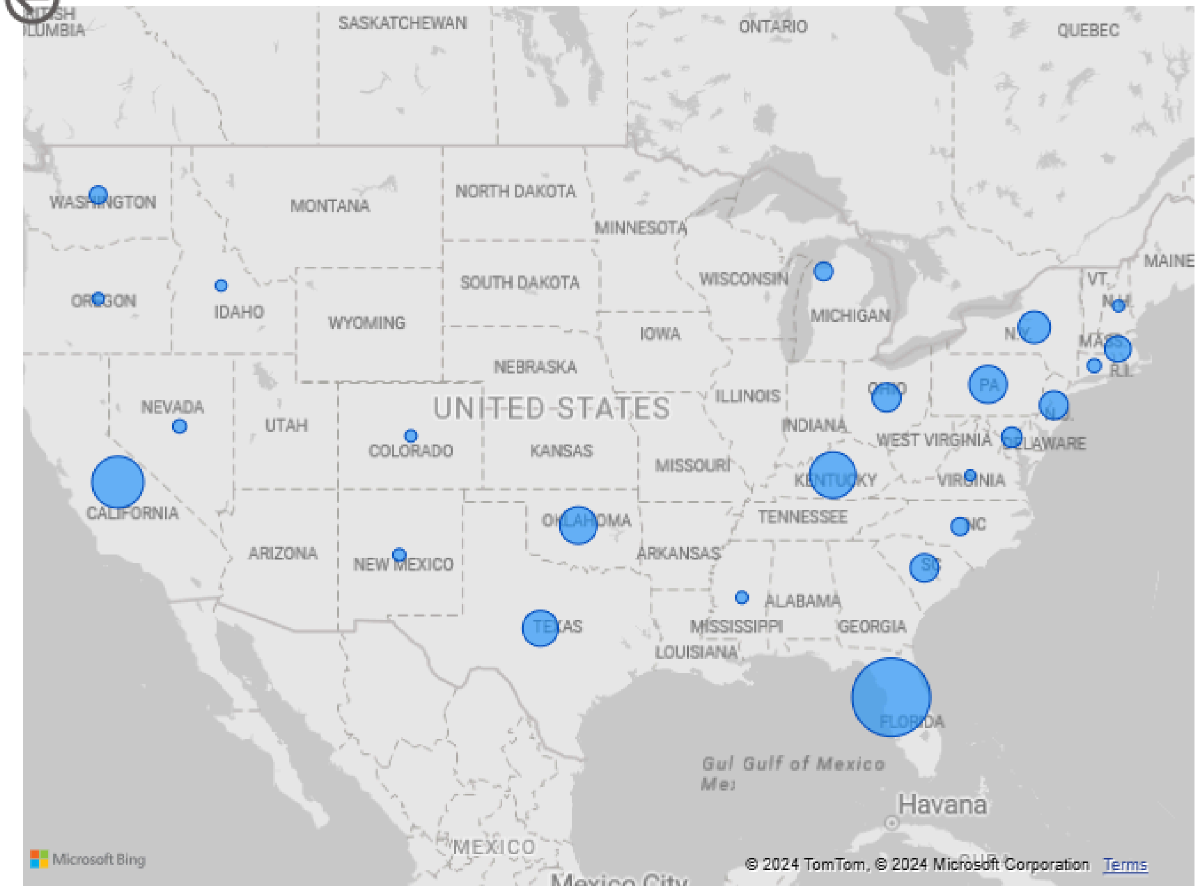
8. Add semantic model refresh activity to pipeline



9. Create a report in Power BI using views dimensions and measures



Total Beneficiaries by State of Prescriber



Generic Name of Drug

□ (Blank)

□ Abacavir/Dolutegravir/Lamivudi

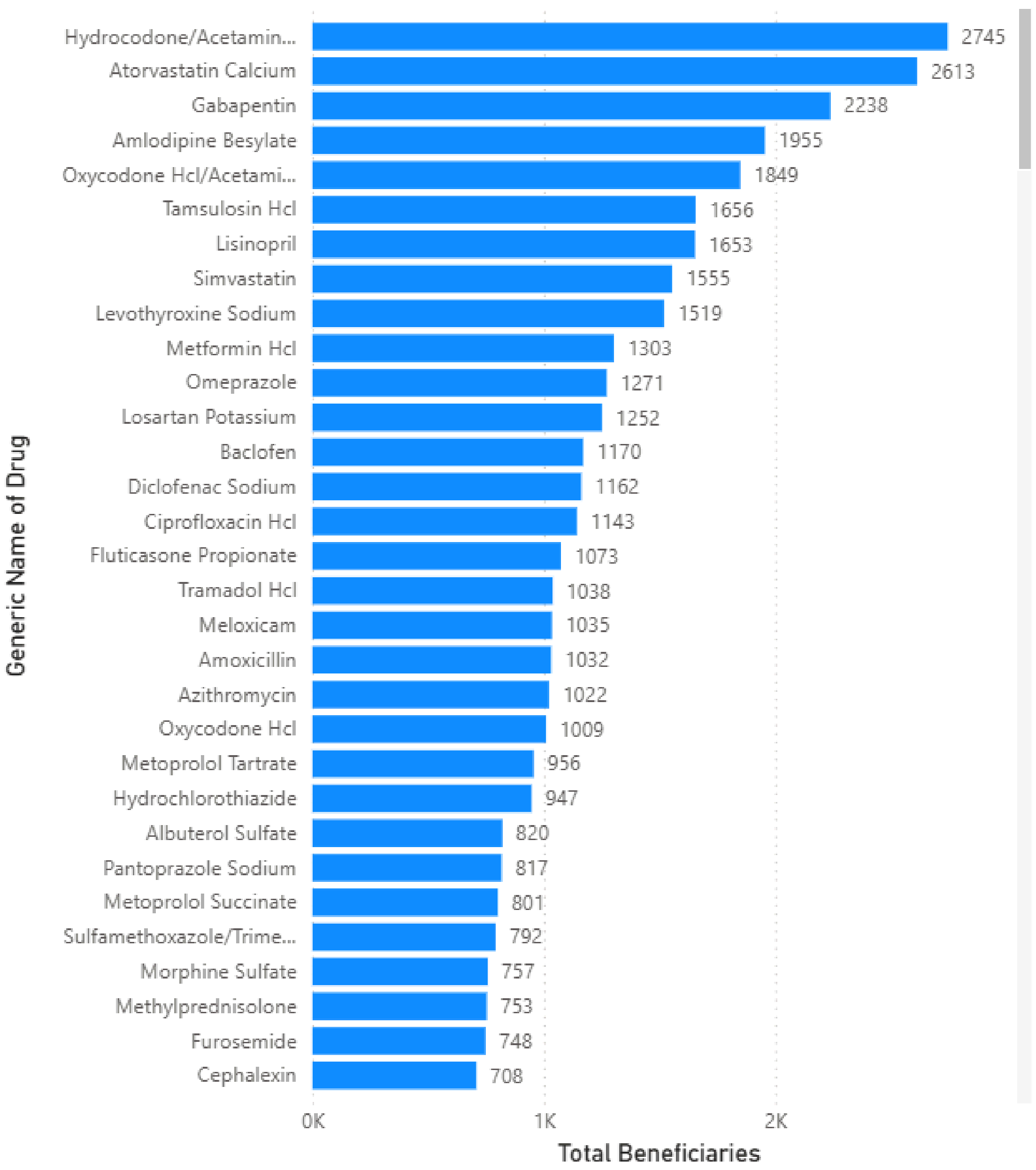
□ Acetaminophen With Codeine

□ Acyclovir

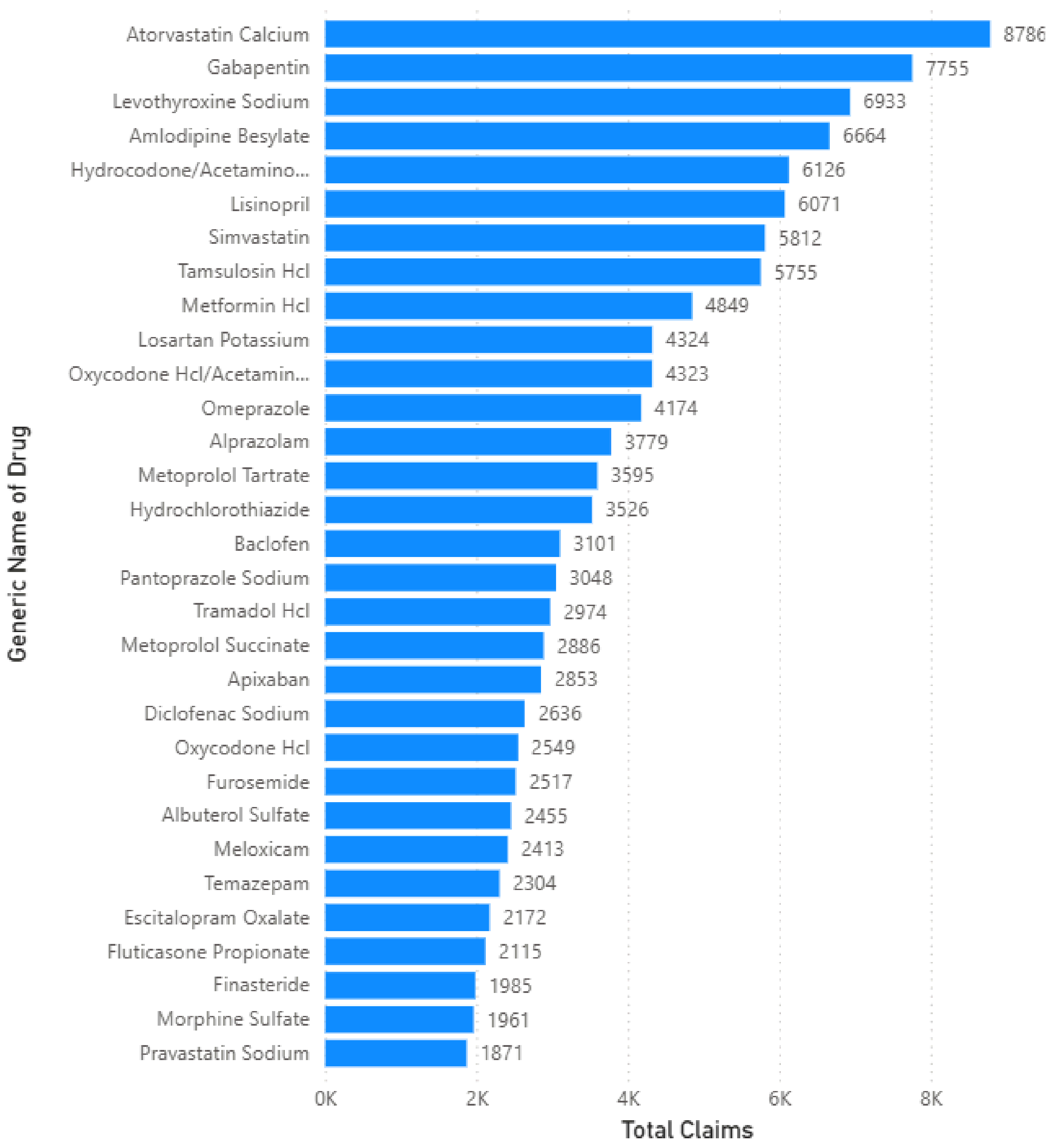
□ Adalimumab

□ Albuterol Sulfate

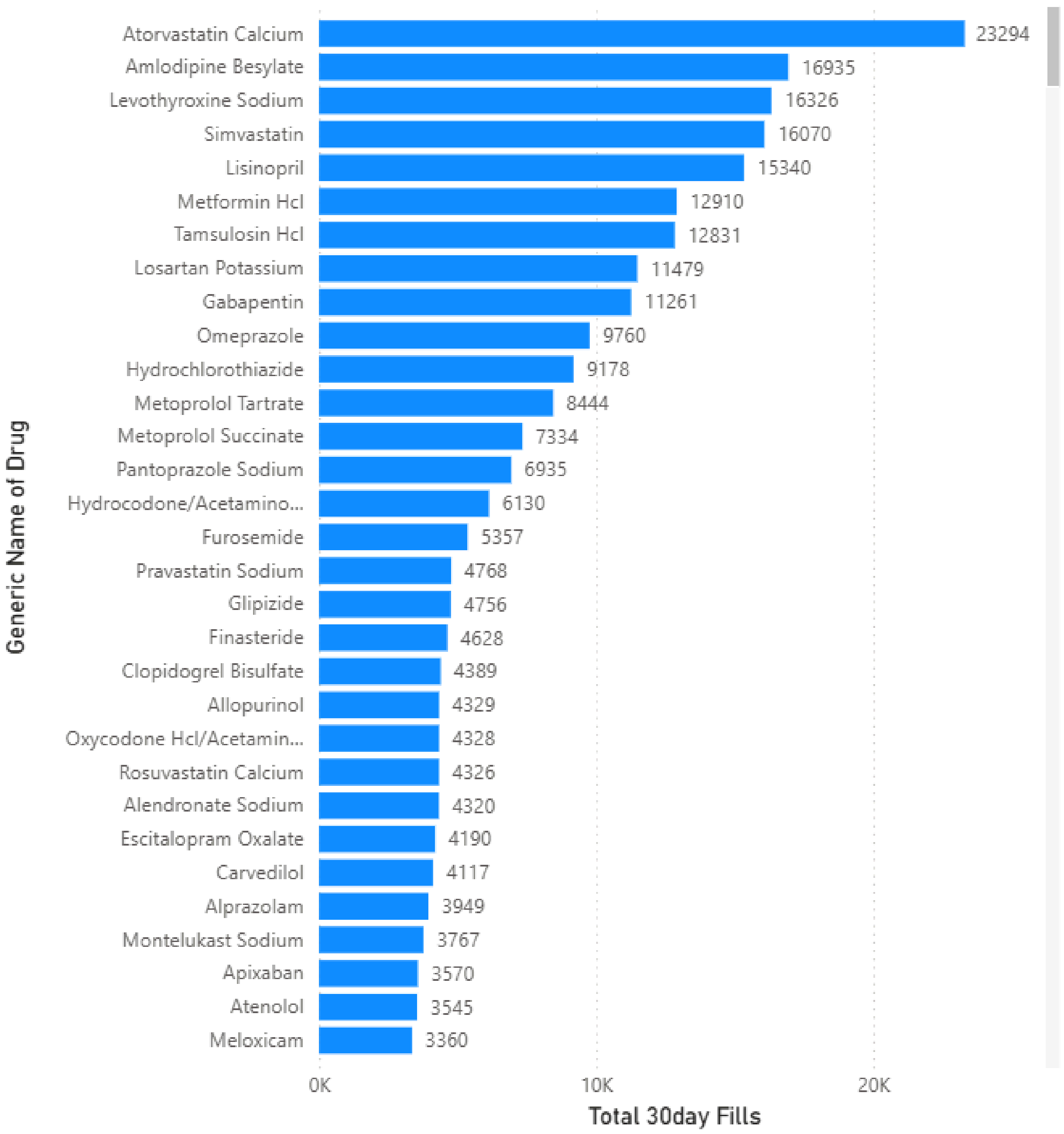
Total Beneficiaries by Generic Name of Drug



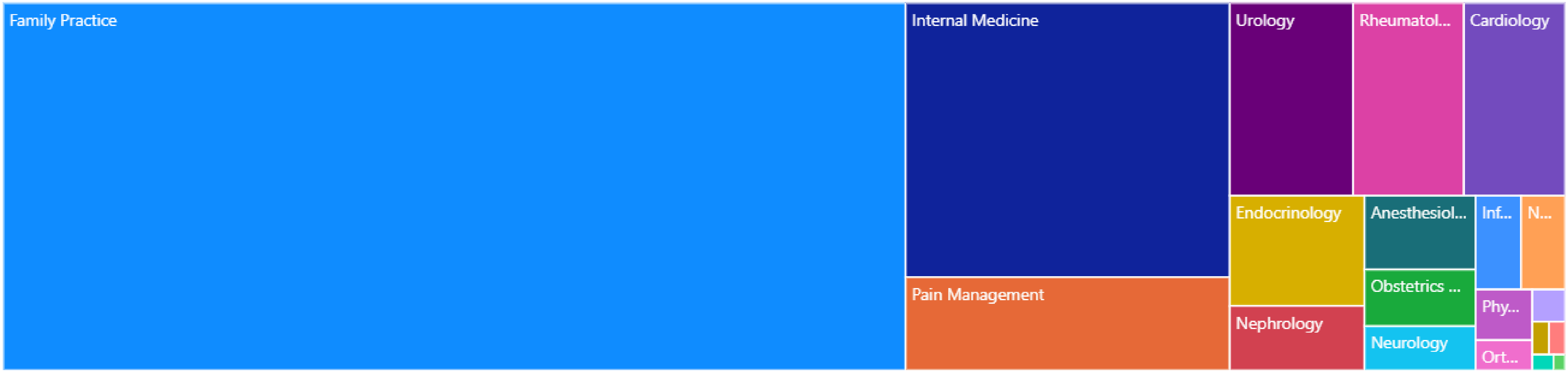
Total Claims by Generic Name of Drug



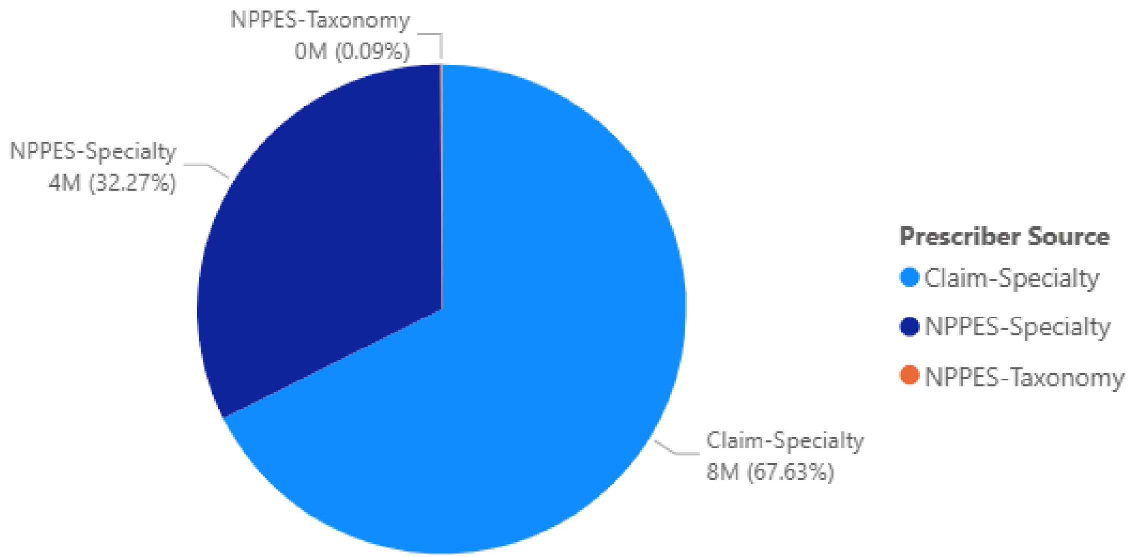
Total 30day Fills by Generic Name of Drug



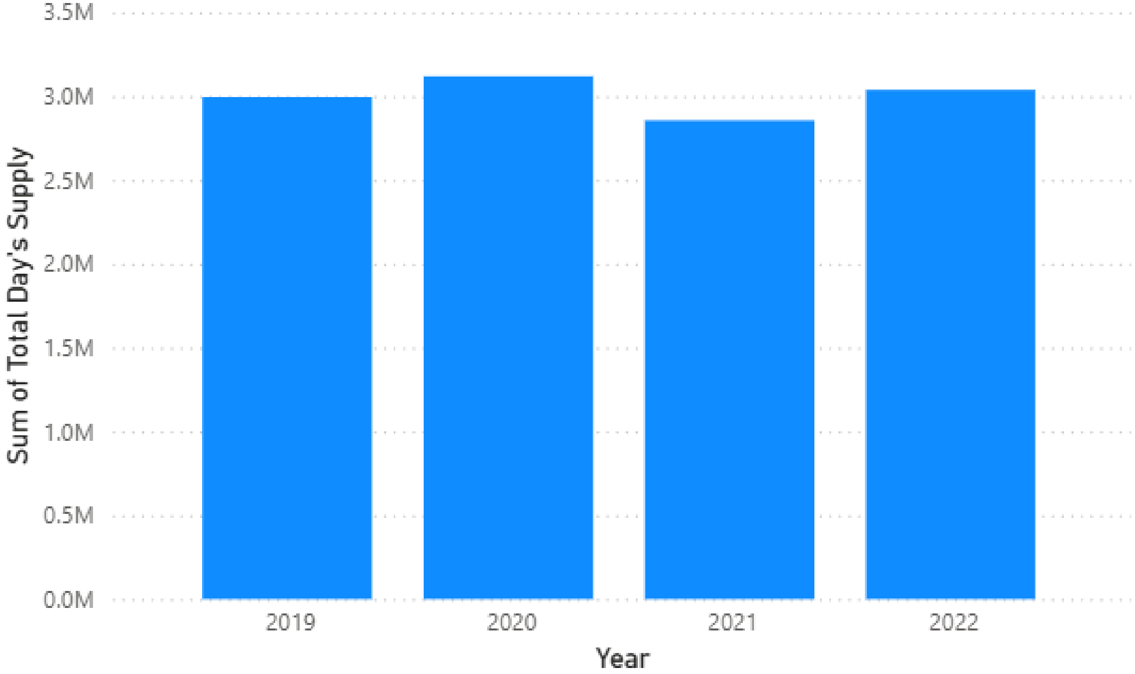
Total Day's Supply by Prescriber Type



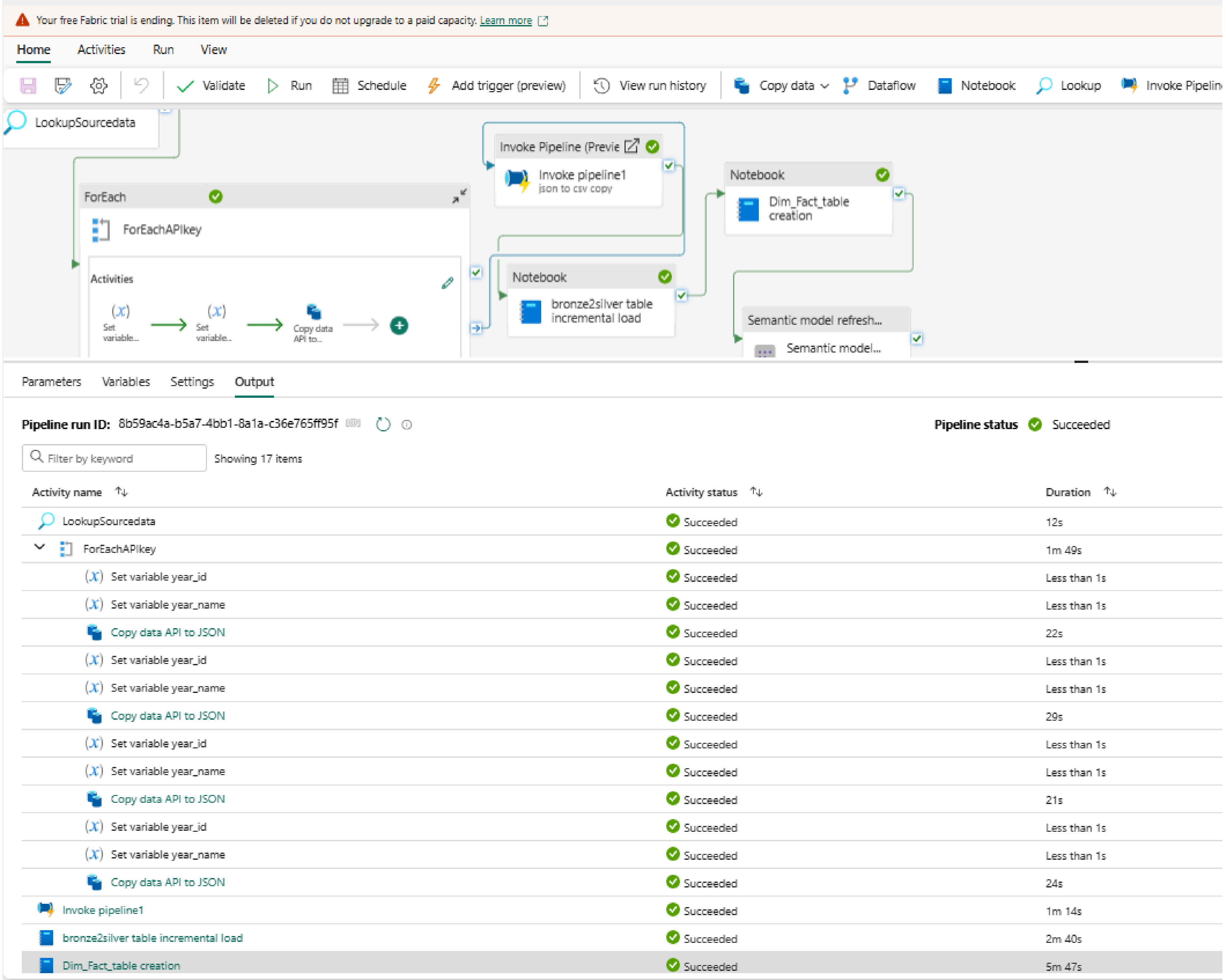
Total Day's Supply by Prescriber Source



Sum of Total Day's Supply by Year



10. Test Pipeline Run



Your free Fabric trial is ending. This item will be deleted if you do not upgrade to a paid capacity. Learn more

Pipeline run ID: 8b59ac4a-b5a7-4bb1-8a1a-c36e765ff95f

Pipeline status ✓ Succeeded

Showing 17 items

Activity name	Activity status	Duration
LookupSourcedata	✓ Succeeded	12s
ForEachAPIkey	✓ Succeeded	1m 49s
Set variable year_id	✓ Succeeded	Less than 1s
Set variable year_name	✓ Succeeded	Less than 1s
Copy data API to JSON	✓ Succeeded	22s
Set variable year_id	✓ Succeeded	Less than 1s
Set variable year_name	✓ Succeeded	Less than 1s
Copy data API to JSON	✓ Succeeded	29s
Set variable year_id	✓ Succeeded	Less than 1s
Set variable year_name	✓ Succeeded	Less than 1s
Copy data API to JSON	✓ Succeeded	21s
Set variable year_id	✓ Succeeded	Less than 1s
Set variable year_name	✓ Succeeded	Less than 1s
Copy data API to JSON	✓ Succeeded	24s
Invoke pipeline1	✓ Succeeded	1m 14s
bronze2silver table incremental load	✓ Succeeded	2m 40s
Dim_Fact_table creation	✓ Succeeded	5m 47s

Thanks for reading