Data Architecture Design for Global Superstore Dataset

DAMG7370 Final Project - Group 6

Team Members:
Nikhila Devi Maddala 002878220
Sai Puja Kamatam 002818267
Likhitha Velagapudi 002727757

Roles and Responsibilities

Nikhila Devi Maddala - Azure till silver stage

Sai Pujitha Kamatam - Azure till gold stage, connecting to Power BI

Likhitha Velagapudi - Power BI

Objective

Project Objective:
To build a scalable, end-to-end data pipeline on Microsoft Azure to ingest, transform, and visualize the Global Superstore dataset, delivering actionable retail insights through interactive dashboards.

Project Summary:

Dataset: Global Superstore (sales, customers, products, returns) from Kaggle.

Architecture: Leverages Azure Blob Storage, Data Factory, Databricks, Synapse Analytics, and Power BI.

Process:

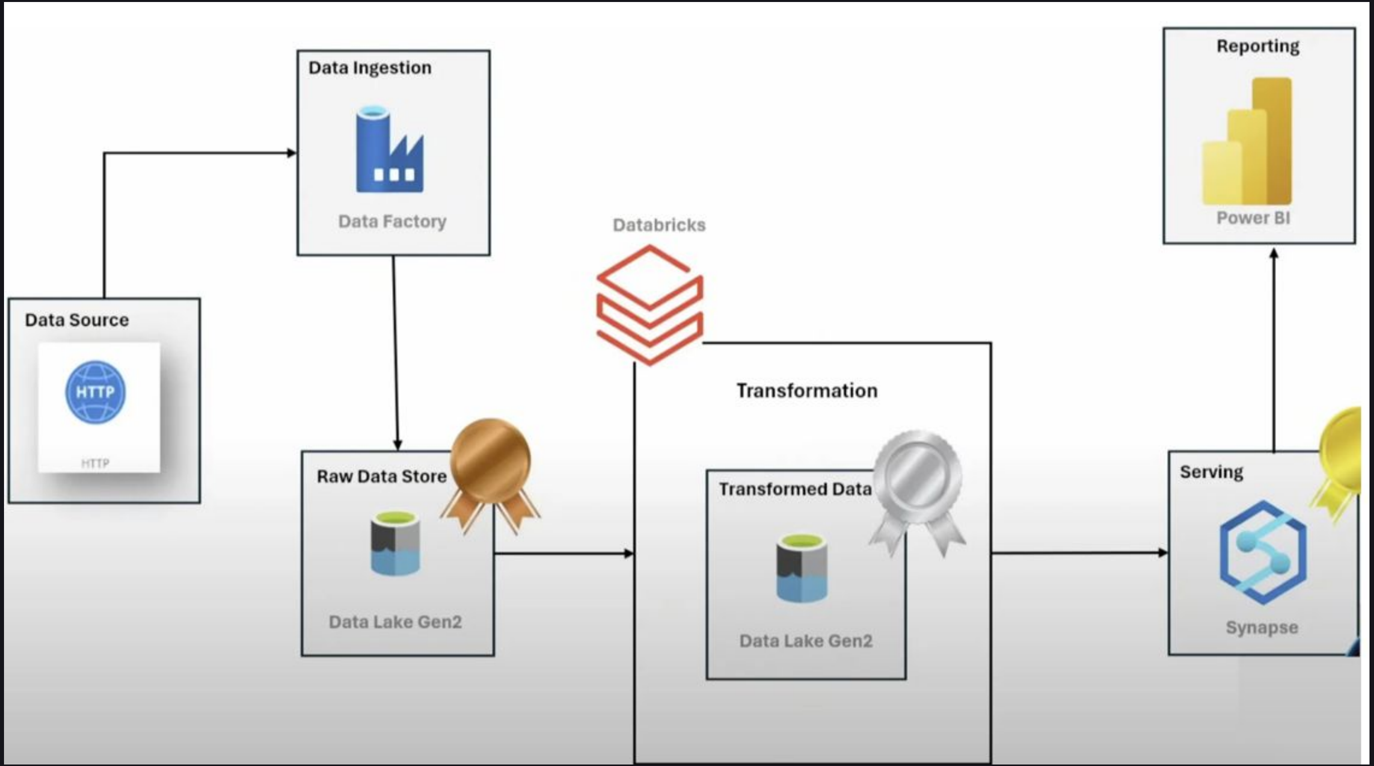
- **Ingestion**: Raw data to Bronze layer via Data Factory.
- **Transformation**: Advanced processing in Databricks to Silver layer.
- **Analytics**: Gold layer views in Synapse for optimized querying.
- **Visualization**: Interactive Power BI dashboards for sales and customer insights.

Outcome: Efficient, scalable pipeline enabling real-time retail analytics.

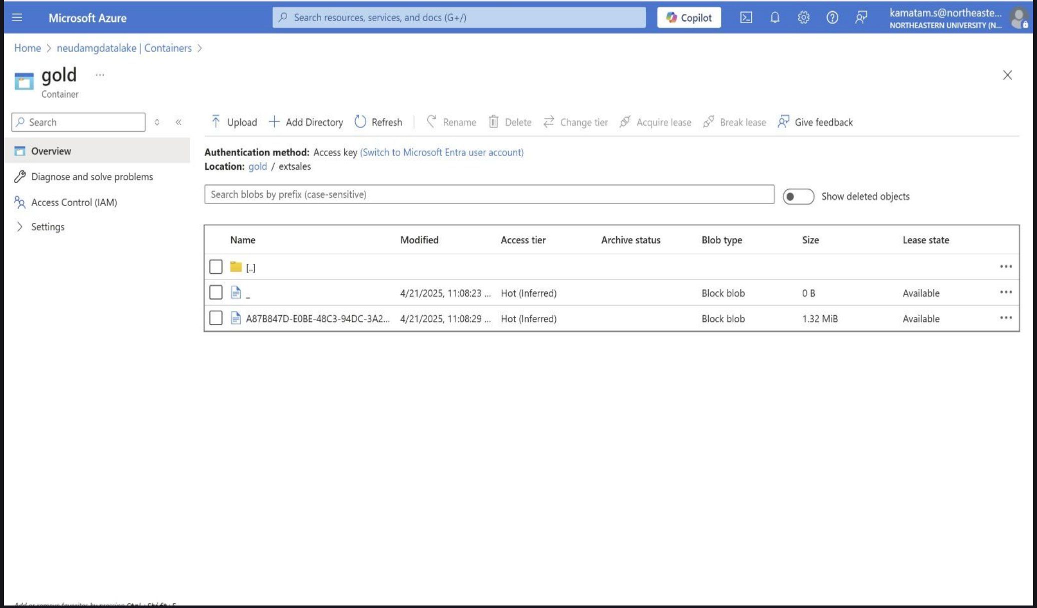
Tools Used in the Global Superstore Data Pipeline

- **Azure Blob Storage / Data Lake Storage Gen2**
  - Stored raw (Bronze), transformed (Silver), and analytics-ready (Gold) data.
  - Enabled hierarchical namespace for efficient file management.
- **Azure Data Factory**
  - Orchestrated data ingestion from Kaggle to Bronze layer.
  - Used static and dynamic pipelines for scalable data copying.
- **Azure Databricks**
  - Performed advanced data transformations (e.g., calculating Shipping Delay Days).
  - Processed data into Parquet files for the Silver layer.
- **Azure Synapse Analytics**
  - Created Gold layer views (e.g., sales, customers) for optimized querying.
  - Supported SQL-based analytics and external tables.
- **Power BI**
  - Built interactive dashboards for sales and customer insights.
  - Connected to Synapse via DirectQuery for real-time visualization.

Data Architecture



Data Source – Global Superstore Dataset



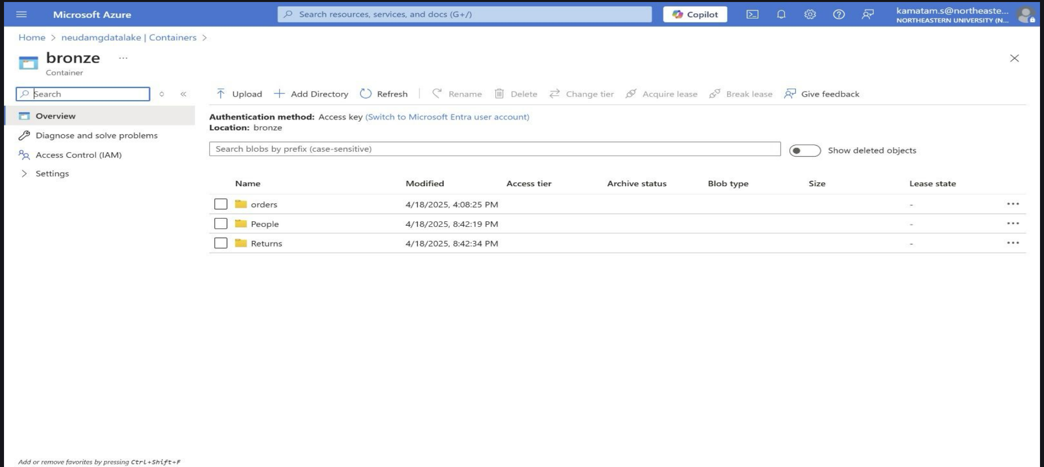
Authentication method: Access key (Switch to Microsoft Entra user account)
Location: gold / extsales

	Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
□	[..]						
□	_	4/21/2025, 11:08:23 ...	Hot (Inferred)		Block blob	0 B	Available
□	A87B847D-E0BE-48C3-94DC-3A2...	4/21/2025, 11:08:29 ...	Hot (Inferred)		Block blob	1.32 MiB	Available

Content:

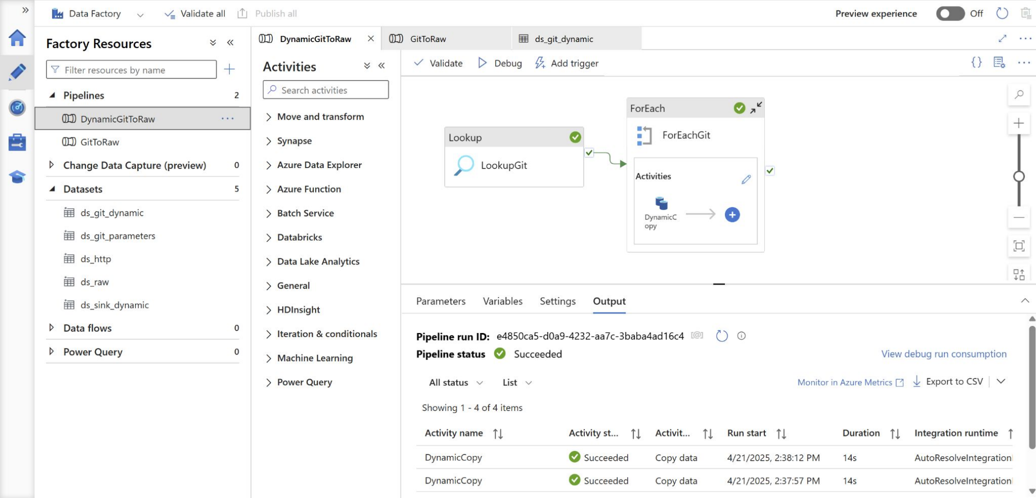
- **Source**: Kaggle Dataset
  - URL: https://www.kaggle.com/datasets/apoorvaappz/global-super-store-dataset
- **Format**: Multiple CSV files
  - Files: Orders, Returns, People
- **Size**: 12.09 MB
- **Upload Method**:
  - Downloaded locally from Kaggle then uploaded to github ..
  - Uploaded to Azure Blob Storage (Bronze container) via Azure Data Factory.
- **Description**: Contains sales, customer, product, and returns data for a global retail business, ideal for analyzing sales trends and operational insights.

Data Ingestion Layer


Microsoft Azure
Search resources, services, and docs (G+/)
Copilot
kamatam.s@northeaste...
NORTHEASTERN UNIVERSITY (N...
Home > neudamgdatalake | Containers >
bronze
Container
Search
Overview
Diagnose and solve problems
Access Control (IAM)
Settings
Upload
Add Directory
Refresh
Rename
Delete
Change tier
Acquire lease
Break lease
Give feedback
Search blobs by prefix (case-sensitive)
Show deleted objects
Add or remove favorites by pressing Ctrl+Shift+F

Authentication method: Access key (Switch to Microsoft Entra user account)
Location: bronze

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
orders	4/18/2025, 4:08:25 PM					-
People	4/18/2025, 8:42:19 PM					-
Returns	4/18/2025, 8:42:34 PM					-



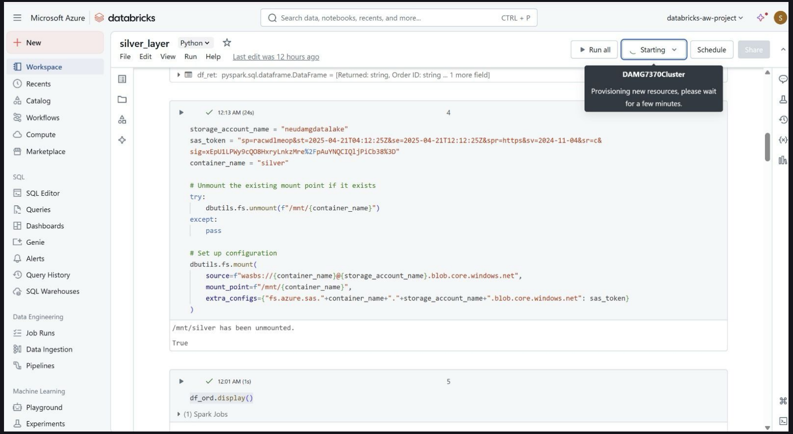
Pipeline run ID: e4850ca5-d0a9-4232-aa7c-3baba4ad16c4
Pipeline status Succeeded

Showing 1 - 4 of 4 items

Activity name	Activity st...	Activit...	Run start	Duration	Integration runtime
DynamicCopy	Succeeded	Copy data	4/21/2025, 2:38:12 PM	14s	AutoResolveIntegration
DynamicCopy	Succeeded	Copy data	4/21/2025, 2:37:57 PM	14s	AutoResolveIntegration

- **Tool**: Azure Data Factory
  - Instance: neu-aw-project (Version 2, East US region).
- **Source**: Kaggle Dataset (Global Superstore)
  - Files: Orders, Returns, People (CSV, 12.09 MB).
  - Downloaded locally for upload.
- **Destination**: Azure Blob Storage (Bronze Container)
  - Storage Account: neudamgdatalake.
  - Stored raw, untouched data for downstream processing.
- **Process**:
  - Created a static pipeline to copy CSV files to Bronze container.
  - Configured dynamic pipelines for scalability (iterative file copying).
- **Outcome**: Reliable, scalable ingestion of raw retail data into the Bronze layer.

Data Cleaning and Transformation- Azure Databricks



DAMG7370Cluster

Provisioning new resources, please wait for a few minutes.

df_ret: pyspark.sql.dataframe.DataFrame = [Returned: string, Order ID: string ... 1 more field]

12:13 AM (24s) 4

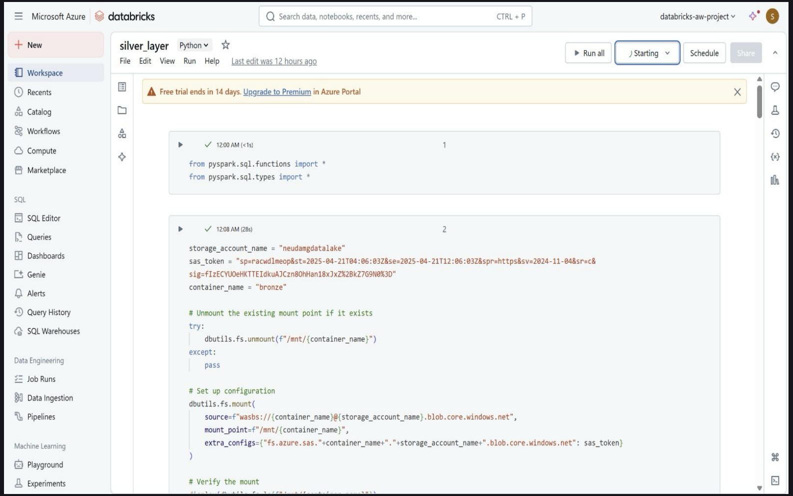
```python
storage_account_name = "neudamgdatalake"
sas_token = "sp=racwdlmeop&st=2025-04-21T04:12:25Z&se=2025-04-21T12:12:25Z&spr=https&sv=2024-11-04&sr=c&
sig=xEpU1LPWy9cQO8HxryLnkzMre%2FpAuYNQCIQljPiCb38%3D"
container_name = "silver"

# Unmount the existing mount point if it exists
try:
    dbutils.fs.unmount(f"/mnt/{container_name}")
except:
    pass

# Set up configuration
dbutils.fs.mount(
    source=f"wasbs://{container_name}@{storage_account_name}.blob.core.windows.net",
    mount_point=f"/mnt/{container_name}",
    extra_configs={"fs.azure.sas."+container_name+"."+storage_account_name+".blob.core.windows.net": sas_token}
)
```

```
/mnt/silver has been unmounted.
True
```

12:01 AM (1s) 5

```python
df_ord.display()
```

(1) Spark Jobs



Free trial ends in 14 days. Upgrade to Premium in Azure Portal

12:00 AM (<1s) 1

```python
from pyspark.sql.functions import *
from pyspark.sql.types import *
```

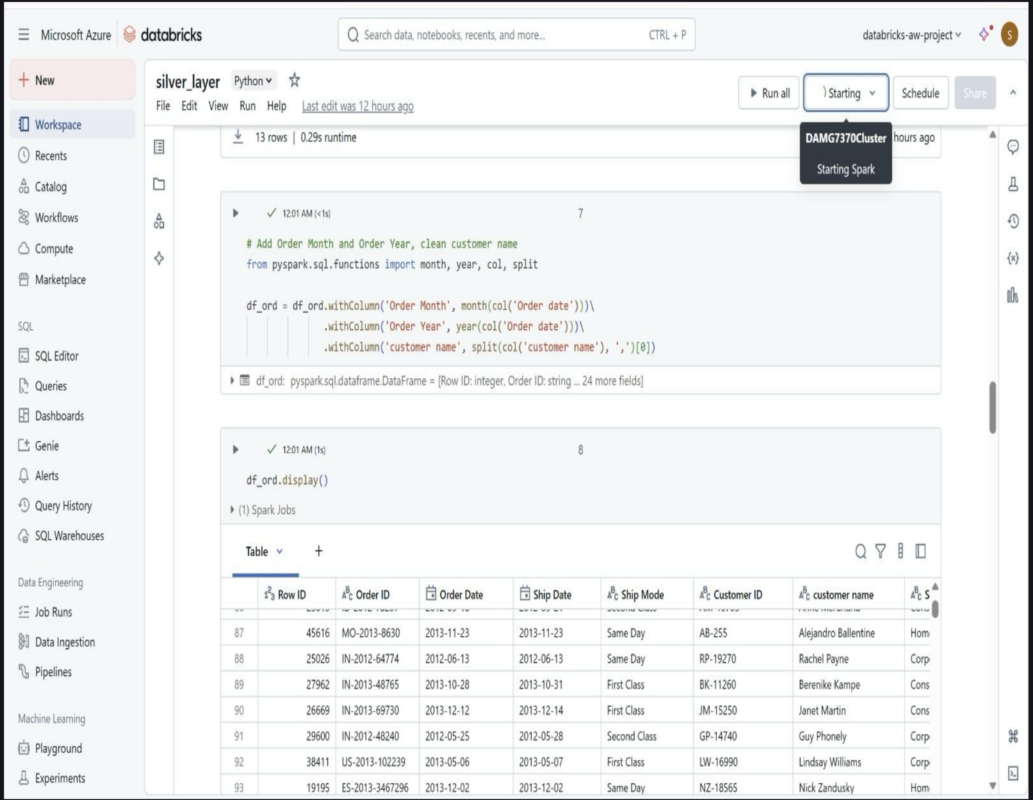
12:08 AM (28s) 2

```python
storage_account_name = "neudamgdatalake"
sas_token = "sp=racwdlmeop&st=2025-04-21T04:06:03Z&se=2025-04-21T12:06:03Z&spr=https&sv=2024-11-04&sr=c&
sig=fIzECYUOeHKTTEIdkuAJCzn8OhHan18xJxZ%2BkZ7G9N0%3D"
container_name = "bronze"

# Unmount the existing mount point if it exists
try:
    dbutils.fs.unmount(f"/mnt/{container_name}")
except:
    pass

# Set up configuration
dbutils.fs.mount(
    source=f"wasbs://{container_name}@{storage_account_name}.blob.core.windows.net",
    mount_point=f"/mnt/{container_name}",
    extra_configs={"fs.azure.sas."+container_name+"."+storage_account_name+".blob.core.windows.net": sas_token}
)

# Verify the mount
```

- **Tool**: Azure Databricks
  - Workspace configured in resource group DAMG7370.
  - Processed data using Spark DataFrames.
- **Input**: Bronze Container (Azure Blob Storage)
  - CSV files: Orders, Returns, People (12.09 MB).
- **Transformations**:
  - **Orders**: Added Order Month, Order Year from Order Date; calculated Shipping Delay Days (Ship Date - Order Date).
  - **Returns**: Standardized Return Reason for consistency.
  - **People**: Unified Region names (e.g., "US-East" to "East US").
- **Output**: Silver Container
  - Saved transformed DataFrames as Parquet files for optimized analytics.
- **Outcome**: Clean, structured data ready for analytics in the Gold layer

Data Cleaning and Transformation- Azure Databricks

- Raw CSV data loaded into PySpark DataFrames from bronze container.
- Silver transformations: added Order Month, Order Year, Shipping Delay Days; standardized Market, region; validated Order ID.
- Transformed data saved as optimized Parquet in the silver container.
- Gold layer views created: calendar, customers, products, returns, sales, subcat, territories.
- Ensures schema consistency and query performance for analysis.



```
# Add Order Month and Order Year, clean customer name
from pyspark.sql.functions import month, year, col, split

df_ord = df_ord.withColumn('Order Month', month(col('Order date')))\
                .withColumn('Order Year', year(col('Order date')))\
                .withColumn('customer name', split(col('customer name'), ',')[0])
```

```
df_ord:  pyspark.sql.dataframe.DataFrame = [Row ID: integer, Order ID: string ... 24 more fields]
```

```
df_ord.display()
```

```
(1) Spark Jobs
```

	Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	customer name	S
87	45616	MO-2013-8630	2013-11-23	2013-11-23	Same Day	AB-255	Alejandro Ballentine	Hom
88	25026	IN-2012-64774	2012-06-13	2012-06-13	Same Day	RP-19270	Rachel Payne	Corp
89	27962	IN-2013-48765	2013-10-28	2013-10-31	First Class	BK-11260	Berenike Kampe	Cons
90	26669	IN-2013-69730	2013-12-12	2013-12-14	First Class	JM-15250	Janet Martin	Cons
91	29600	IN-2012-48240	2012-05-25	2012-05-28	Second Class	GP-14740	Guy Phonely	Corp
92	38411	US-2013-102239	2013-05-06	2013-05-07	First Class	LW-16990	Lindsay Williams	Corp
93	19195	ES-2013-3467296	2013-12-02	2013-12-02	Same Day	NZ-18565	Nick Zandusky	Hom

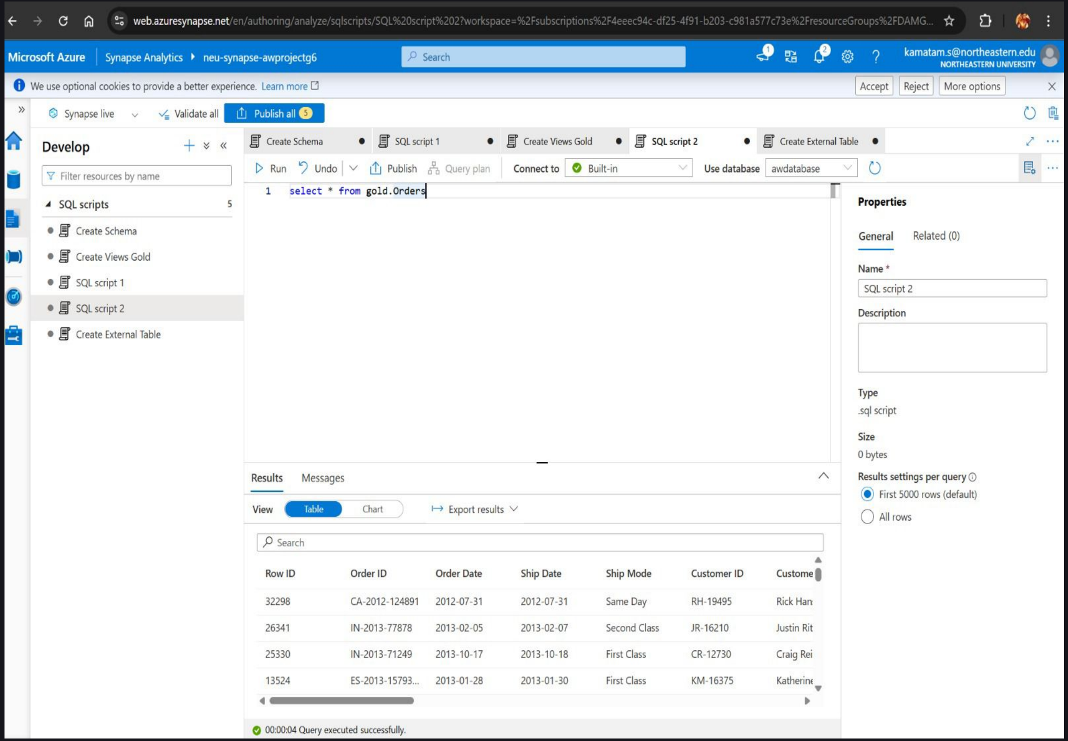
Analytics and Query Layer - Azure Synapse

Tools used: Azure Synapse Serverless SQL

Data Source: External data source to silver container

Views Created:

- Gold.orders
- Gold.sales



```
select * from gold.Orders
```

Name *: SQL script 2

Description

Type: .sql script

Size: 0 bytes

Results settings per query

◉ First 5000 rows (default)

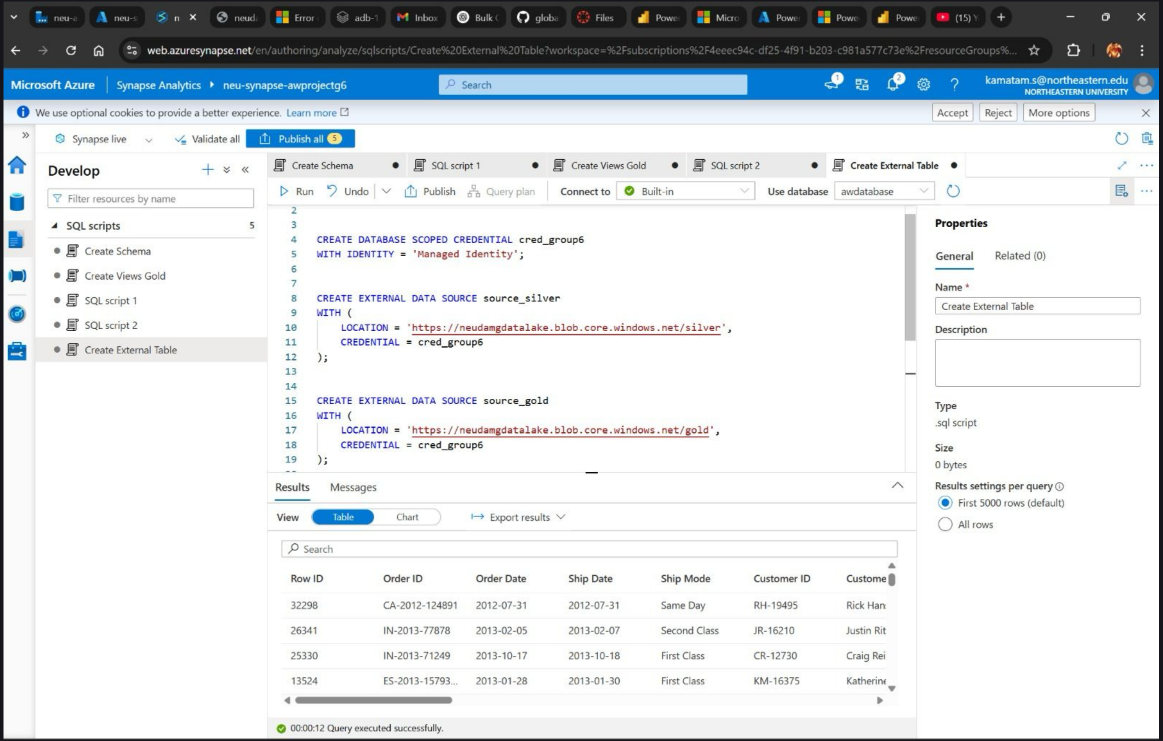
○ All rows

Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Custome
32298	CA-2012-124891	2012-07-31	2012-07-31	Same Day	RH-19495	Rick Han
26341	IN-2013-77878	2013-02-05	2013-02-07	Second Class	JR-16210	Justin Rit
25330	IN-2013-71249	2013-10-17	2013-10-18	First Class	CR-12730	Craig Rei
13524	ES-2013-15793...	2013-01-28	2013-01-30	First Class	KM-16375	Katherine

00:00:04 Query executed successfully.

Analytics and Query Layer - Azure Synapse

- Utilized Managed Identity for secure data access
- SQL queries can be executed directly on silver or gold layer data without needing to copy or move it.
- Azure Synapse supports high-performance operations like joins, aggregations, and filtering on large datasets.



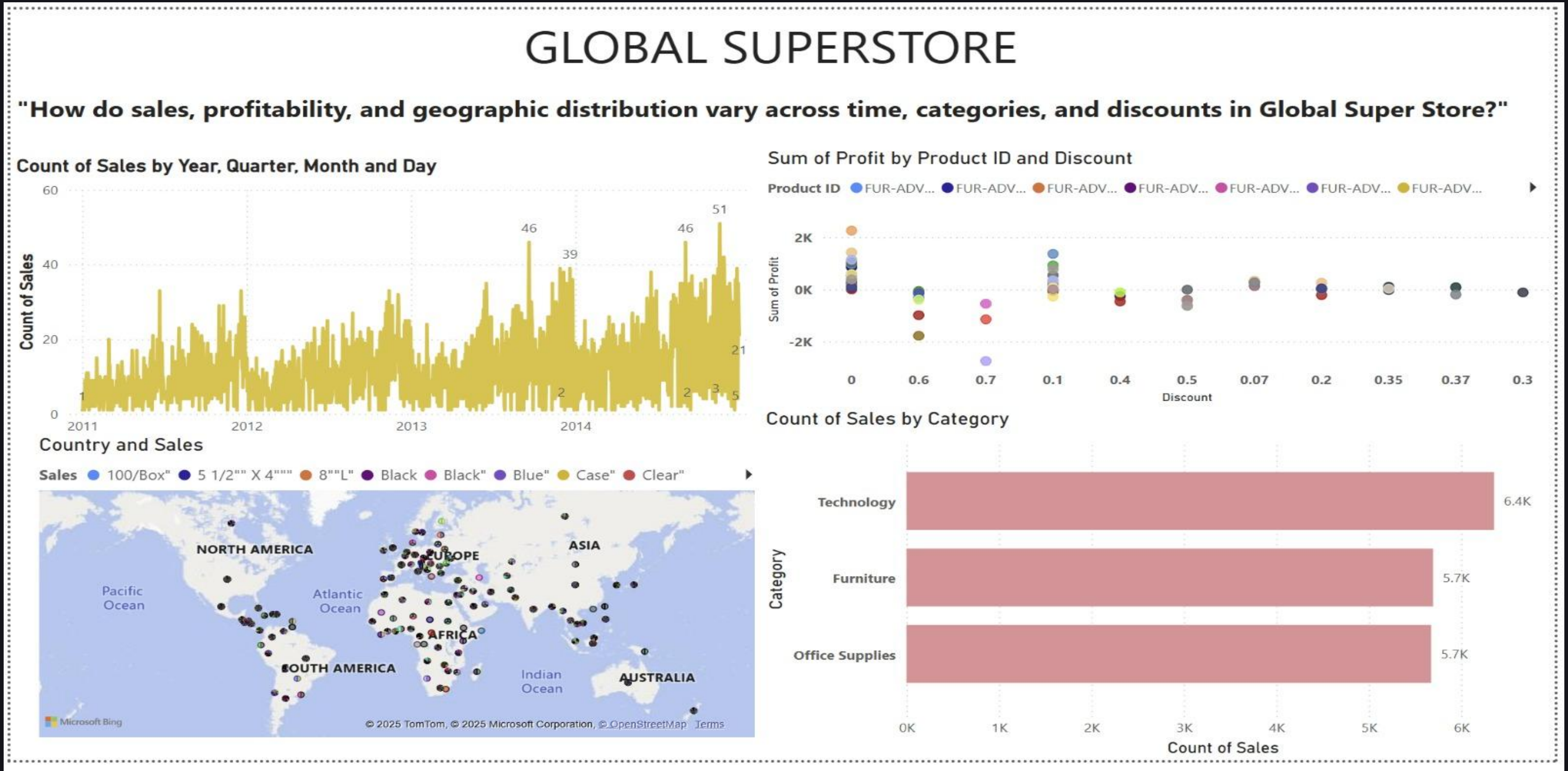
```sql


CREATE DATABASE SCOPED CREDENTIAL cred_group6
WITH IDENTITY = 'Managed Identity';


CREATE EXTERNAL DATA SOURCE source_silver
WITH (
    LOCATION = 'https://neudamgdatalake.blob.core.windows.net/silver',
    CREDENTIAL = cred_group6
);


CREATE EXTERNAL DATA SOURCE source_gold
WITH (
    LOCATION = 'https://neudamgdatalake.blob.core.windows.net/gold',
    CREDENTIAL = cred_group6
);
```

Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Custome
32298	CA-2012-124891	2012-07-31	2012-07-31	Same Day	RH-19495	Rick Han
26341	IN-2013-77878	2013-02-05	2013-02-07	Second Class	JR-16210	Justin Rit
25330	IN-2013-71249	2013-10-17	2013-10-18	First Class	CR-12730	Craig Rei
13524	ES-2013-15793...	2013-01-28	2013-01-30	First Class	KM-16375	Katherine

00:00:12 Query executed successfully.

Power BI Dashboard

Visualizations created:



GLOBAL SUPERSTORE

"How do sales, profitability, and geographic distribution vary across time, categories, and discounts in Global Super Store?"

Count of Sales by Year, Quarter, Month and Day

Sum of Profit by Product ID and Discount

Country and Sales

Count of Sales by Category

THANK YOU