**Data Pipeline for Customer Account Analysis**

**Objective:**

This project aimed to design and implement a data pipeline to process customer account data efficiently. The pipeline ingested data from a backend storage account, transformed it using Databricks, and upserted the refined data into a dedicated SQL pool in Azure Synapse Analytics. The pipeline ensures scalability, accuracy, and efficient data processing for downstream analytics.

**Backend Data:**

Data source for this project is from backend blob storage container and HTTP(GitHub repo) as below

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

AllHTTPfiles.json – in blob storage has connection details for http source, which will be fetched on fly.

**Customer Account Workspace (Resource Group):**

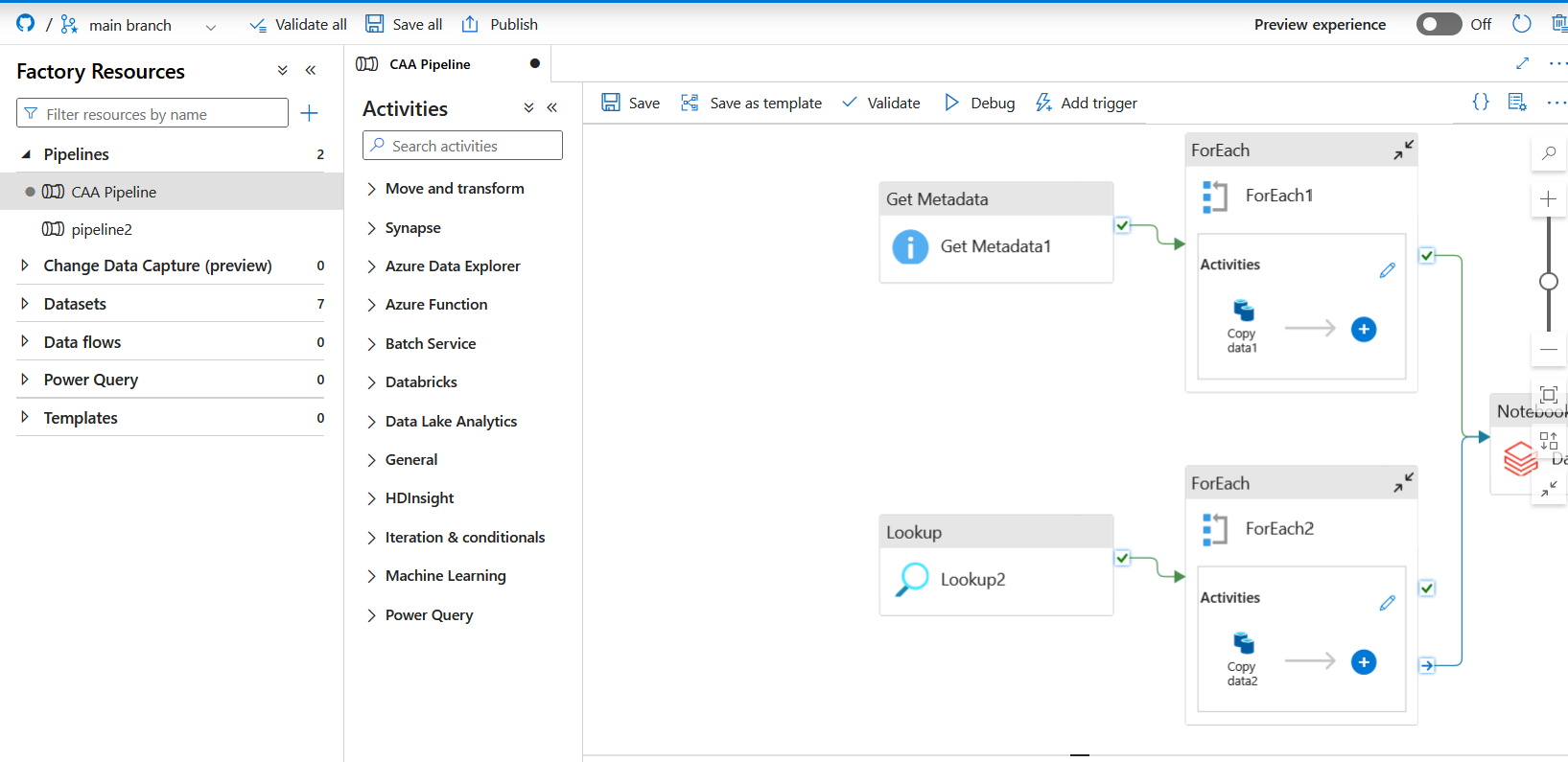
Holds necessary configured resources for this project as below

A screenshot of a computer

AI-generated content may be incorrect.

**Step 1 – (Ingesting data from raw to bronze)**

Purpose of this step is to get data from raw data sources to bronze layer as its raw form. Get Metadata1 activity will fetch list of files from backend adls container and forEach1 will loop through each file from get metadata1 activity and copy activity inside forEach will copy each file to bronze layer in its raw form. Lookup2 will read AllHTTPfiles.json in backend raw container, which holds necessary connection details for other 2 files in GitHub and ForEach2 will loop through those files and copy files into bronze layer in its raw form.



**Service Principal:**

Caa\_app service principal is created and assigned role of Storage Blob data contributor to both backednd adls and bronze layer adlsgen2 storage account.

A screenshot of a computer

AI-generated content may be incorrect.

**Azure Key Vault:**

A screenshot of a computer

AI-generated content may be incorrect.

As part of this process, linked services and datasets were created as needed. For linked services, authentication type Service Principal is utilized for better security and Azure Key vault for storing service Principal Secrets.

A screenshot of a computer

AI-generated content may be incorrect.

**Datasets:**

Necessary datasets for this process are created leveraging linked services as below.

A screenshot of a computer

AI-generated content may be incorrect.

**Bronze container:**

As expected after a pipeline run, all raw data from both data sources are ingested into bronze layer.

A screenshot of a computer

AI-generated content may be incorrect.

**Step 2: Data Cleaning (Bronze to Silver)**

As part of this process, Databricks workspace – CAA-workspace was configured and given access to data factory through access token which is managed as part of key vault storage. Linked service ls\_caa\_databricks is created to secure the access between databricks and ADF.

**A screenshot of a computer

AI-generated content may be incorrect.**

Mounting of bronze container (source) is done to mounting point /mnt/bronze and silver container (Sink) is done to mounting point /mnt/silver using service principal. As part of this Secret Scope named - CAA\_Secret\_Scope is created and utilized.

Read raw data from the Bronze Layer. Performed data cleansing (removed duplicates, handled missing values). Applied schema enforcement (avoided inferSchema for data consistency). Saved cleaned data in the Silver Layer (Curated Container) in Parquet format. While reading data from bronze only new file which uploaded on that particular day is read as part of this by fetching today’s date in databricks.

**today\_date = datetime.today().strftime("%Y\_%m\_%d")**

**file\_path\_accounts = f"/mnt/bronze/accounts\_{today\_date}.csv"**

****

**Silver Container:**

As expected after a pipeline run, all raw data from bronze layer is cleaned and ingested into silver layer.

A screenshot of a computer

AI-generated content may be incorrect.

**Step 3: Business Transformation (silver to gold)**

Read data from the Silver Layer and applied business logic for aggregations and calculations:

1. **Customer Balance Summary:** Aggregated balances per customer.
2. **Loan Summary:** Average loan amounts grouped by loan term.
3. **Loan Amount & Interest Calculation:** Computed total loan amounts with interest.
4. **Frequent Transaction Type:** Identified the most frequent transaction type.
5. **Transaction Analysis**: Aggregated transactions per account and type.

Saved the transformed data into the Gold Layer (Refined Container) in Delta format.

Mounting of silver container (source) is done to mounting point /mnt/silver and gold container (Sink) is done to mounting point /mnt/gold using service principal. As part of this Secret Scope named - CAA\_Secret\_Scope is created and utilized.



A screenshot of a computer

AI-generated content may be incorrect.

**Gold Layer:**

As expected after a pipeline run, all silver data from transformed aggregated and ingested into gold layer.

A screenshot of a computer

AI-generated content may be incorrect.

**Step 4: Data Warehousing in Synapse**

Dedicated sql pool named – caa\_dw is created for data warehousing and necessary tables are created as below:

**A screenshot of a computer

AI-generated content may be incorrect.**

As part od this , necessary linked service and datasets are created for secure authentication between adlsgen2 and synapse.

Individual copy activity is created for all 5 tables to copy gold layer data into synapse tables respectively. Enabled **Upsert functionality** in sink for every copy activity to accomplish incremental loading functionality.

A screenshot of a computer

AI-generated content may be incorrect.

**Synapse:**

After successful execution of pipeline, all data warehouse tables are populated with gold layer data. Second day data is also passed to check incremental loading process, which is successfully executed.

A screenshot of a computer

AI-generated content may be incorrect.

**Pipeline Diagram:**

A diagram of a person's workflow

AI-generated content may be incorrect.

**Pipeline Execution:**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Key Features Implemented:**

✅ Incremental Processing: Handled using Delta format in Databricks.  
✅ Schema Enforcement: Avoided inferSchema, enforced data types explicitly.  
✅ Aggregations & Business Metrics: Performed meaningful transformations in the Gold layer.  
✅ Dynamic Pipelines: Used ForEach in ADF for efficient Synapse loading.  
✅ Secure Connections: Managed credentials using Azure Key Vault.  
✅ Upsert in Synapse: Configured MERGE for handling updates and inserts.

**GitHub Link:**

https://github.com/Arunkumar-Senthilkumar/Customer-Account-Analysis-ETL