### **Project Requirements: Step-by-Step**

Input Dataset : Medical Claim Dataset given in the link below – 1 Million records

<https://www.kaggle.com/datasets/drscarlat/medicalclaimssynthetic1m>

Ingest the data in On-Prem using Pyspark and the split the data into 6 subsets of .csv files. Upload the 6 input

datasets in ADLS in an interval of 1 hour each to reproduce the incremental data load scenario of hourly.

#### **1. Initial Setup**

* **Source**: Claim dataset (CSV)
* **Initial Split**:
  + Divide dataset into **6 subsets** (e.g., subset\_1.csv to subset\_6.csv)
  + Add md5\_hash column using pandas code
  + Each subset processed **every 1 hour**
  + **Folder Structure**:

stg/ # Raw CSV subsets

preprocess/ # Parquet files

sink/ # Final fact/dimension tables

stg/archive/ # For archived data

preprocess/archive/ # For purged data

sink/archive/ # For archived data

#### **2. Pipeline 1: Data Ingestion & CSV → Parquet**

**Input**: stg/subset\_{N}.csv  
**Output**: preprocess/subset\_{N}.parquet  
**Steps**:

1. Gets triggered when a .csv file falls into /stg folder
2. Read .csv from stg/
3. Convert to Parquet
4. Save to preprocess/ with same structure and parquet file extension

#### **3. Pipeline 2: Parquet → Fact/Dimension Tables**

**Input**: preprocess/subset\_{N}.parquet  
**Output**: Append to single files in sink/:

* sink/dim\_beneficiary.csv
* sink/dim\_provider.csv
* sink/fact\_claim.csv

**Steps**:

1. Gets triggered when a .parquet file falls into /preprocess folder
2. **Read parquet** from latest subset
3. **Generate Surrogate Keys**:

* Start keys from last run’s max value
* Example: If dim\_beneficiary last key was 100, new subset starts at 101
* **Append Data**:
* Append to existing fact and dimension CSV files in sink/

#### **4. Scheduling & Automation**

* **Trigger Pipeline 1** immediately when a new subset arrives in stg/
* **Trigger Pipeline 2** after Pipeline 1 succeeds and when a .parquet file falls in /preprocess folder
* **Frequency**: Every 1 hour (per subset)

#### **6. Data Lifecycle Management**

* **Archiving**:
  + Move processed files from stg/, preprocess/, sink/ to their respective archive/ folders after **1 day**
* **Purging**:
  + Delete files from archive/ after **2 days**