Healthcare ETL Pipeline using Azure Databricks

# 1. Introduction

This document outlines the implementation of an end-to-end ETL (Extract, Transform, Load) pipeline for processing healthcare patient data. The pipeline is built using Azure Databricks and other Azure services, with the goal of enabling predictive analytics and dashboard reporting for healthcare insights.

# 2. Use Cases

- Predicting patient readmission risks

- Monitoring patient diagnosis trends

- Generating regulatory reports

- Feeding cleaned data into dashboards for insights

# 3. Architecture Overview

The architecture includes the following components:  
- Azure Data Lake Storage: stores raw patient data files  
- Azure Databricks: performs data transformation using PySpark  
- Azure SQL Database / Snowflake: stores cleaned and processed data  
- Power BI / Snowsight: builds dashboards for visualization  
- (Optional) Azure Data Factory: orchestrates the pipeline execution

# 4. Requirements

- Azure Subscription with access to Databricks, ADLS, and Azure SQL/Snowflake

- Databricks cluster with PySpark and Snowflake connector installed

- Power BI Desktop or Snowsight for dashboarding

# 5. Step-by-Step Process

1. Set up an Azure Databricks workspace and cluster

2. Upload `patient\_data.csv` into Azure Data Lake or DBFS

3. Create and run a Databricks notebook (`healthcare\_etl\_pipeline.py`) to:

- Read data using PySpark

- Clean and transform data (remove nulls, derive age, parse dates)

- Write cleaned data into Azure SQL or Snowflake

4. Perform optional machine learning (readmission risk prediction)

5. Connect Power BI / Snowsight to the database for dashboard reporting

# 6. Output

- Cleaned patient data stored in Azure SQL or Snowflake

- Dashboards showing:

• Readmission rate by diagnosis

• Monthly visit trends

• Age distribution by gender

- Optional: ML model to predict readmission risks

# 7. Conclusion

This ETL pipeline enables healthcare providers to process and analyze large-scale patient data for decision making and reporting. The modular architecture makes it scalable and easy to integrate with ML and BI tools.