A Data Engineer must

- Build pipelines that serve the business – increase revenue, generate sales, optimize processes, etc.
- Make their Data Analysts' lives easier ☺

🏪 The Grocery Store That Became Data-Driven

Every evening, the manager of GreenLeaf Grocery sits down with a pile of log books.
One shows what was sold that day. Another lists deliveries from suppliers. A third holds notes from customers.

At first, it’s just pages and pages of scribbles — numbers, dates, and comments. But the manager wants to understand what’s really happening in the store. That’s where the **data engineering process** begins.

🏪 The Grocery Store That Became Data-Driven

1. **Ingestion – Gathering the information**
   The manager collects all the log books and receipts from different sections — fruits, dairy, bakery, and suppliers.
   In the data world, this is called *ingestion*: bringing data from many sources into one place.
2. **Transformation – Cleaning and organizing**
   Some pages are smudged, some prices are missing, and the word “tomato” is written five different ways.
   So the manager spends time cleaning things up — fixing errors, making names consistent, and double-checking totals.
   That’s *transformation*: turning messy data into clean, usable information.

 The Grocery Store That Became Data-Driven

3. Storage – Putting things where they belong

Once everything is neat, the manager files it away carefully — one binder for sales, one for suppliers, one for feedback.
In modern terms, this is like storing data in a database or data warehouse, so it's safe and easy to search.

4. Reporting – Learning from the data
Finally, the manager opens a summary sheet:

- Apples sold: 250 kg
- Most popular product: Bread
- Supplier delays: 2 days this week
- That's *reporting*: turning organized data into insights that help make better decisions — like ordering more bread or finding a faster supplier.

Pre-requisites

- No prior knowledge assumed
- Cloud fundamentals would be beneficial, not necessary
- Basic knowledge of SQL would be beneficial, not necessary
- Azure Account – *we will be creating one* ☺

In this course, we will be using

Azure Data Factory – for ingestion, running pipelines, etc.

Azure Data flow – for transformations

Azure Data Lake Storage Gen2 – for storing data for raw and transformed layers

Azure SQL Database – for storing the final data for reporting layer

Power BI – for reporting

Azure Services

Azure Data Factory is a cloud-based, serverless data integration service that orchestrates and automates data movement and transformation for hybrid and cloud-based data solutions. It is used to build and schedule data-driven workflows, known as pipelines, that can ingest data from various sources, transform it, and load it into destinations like data warehouses.

An Azure Data Factory data flow is a visually-designed, code-free data transformation tool that allows you to build complex data manipulation logic without writing code. It runs on scalable, managed Apache Spark clusters and is used as an activity within an Azure Data Factory pipeline to transform data from a source to a sink. Data flows provide a graphical user interface to define transformations such as joins, aggregations, and filters, and they enable interactive debugging with a live preview of the data.

Azure Data Lake Storage Gen2 (ADLS Gen2) is a massively scalable data lake solution built on Azure Blob Storage that is designed for big data analytics and integrates with Azure's analytics services to provide a fast, secure, and cost-effective platform for storing and processing large volumes of structured, semi-structured, and unstructured data.

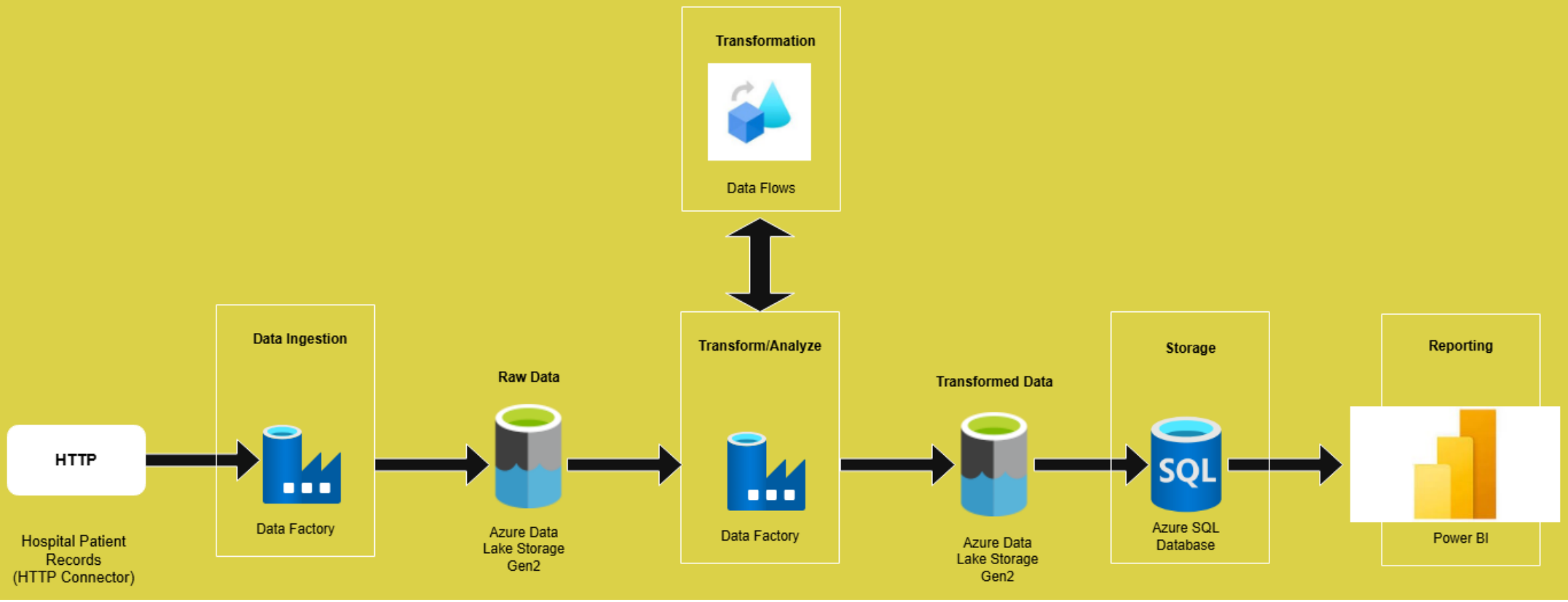
Azure Services

Azure SQL Database is a fully managed Platform-as-a-Service (PaaS) relational database service offered by Microsoft Azure. It is built on the latest stable version of the Microsoft SQL Server database engine and provides a highly available, scalable, and secure data storage layer for cloud applications.

Power BI is a Microsoft business analytics service that lets users connect to various data sources, transform the data, and create interactive reports and dashboards to visualize insights. It is a unified platform for business intelligence that helps organizations make fast, informed decisions by providing tools for data analysis, visualization, and sharing.

Solution Architecture

Hospital Patient Records Data | End-to-End Azure Data Engineering Project



Understanding the data

- Patients
- Payers
- Encounters
- Procedures

Data available at: *https://github.com/snowydata-sarthak/Hospital-Patient-Records/tree/main/data*

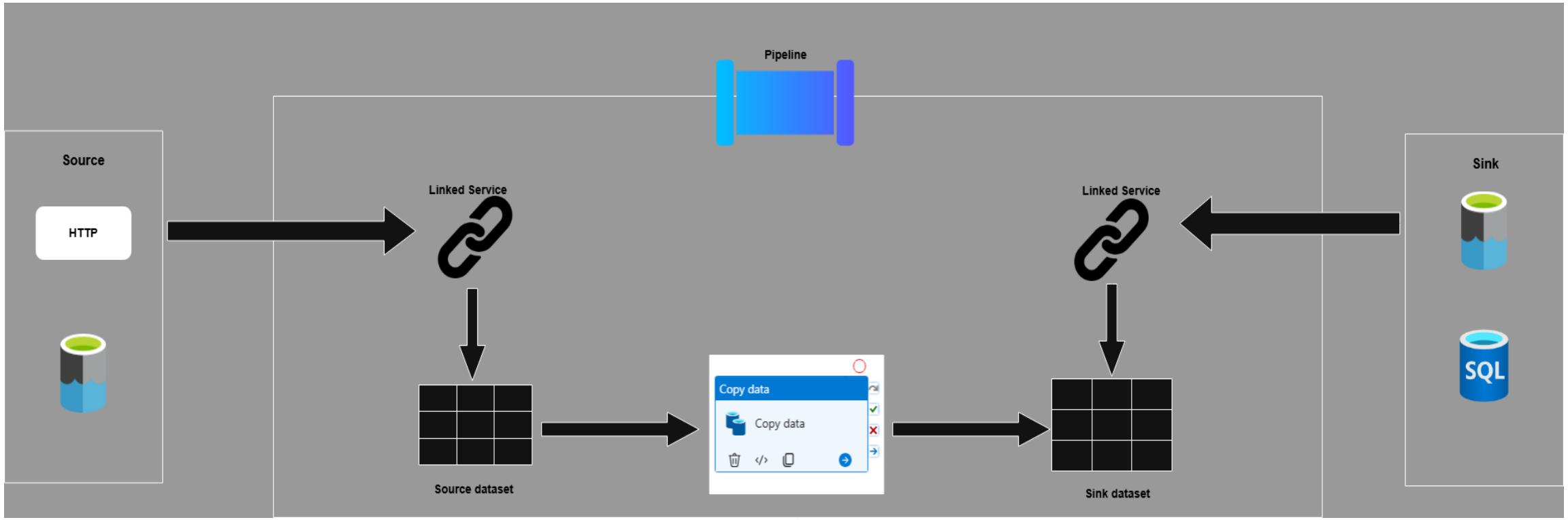
Original data source: *https://mavenanalytics.io/data-playground/hospital-patient-records*

Setting up the Environment

- Azure Subscription
- Data Factory
- Data Lake Storage Gen2
- Azure SQL Database

Data Ingestion

HTTP ⟹ Azure Data Lake



Questions we are trying to answer

OBJECTIVE 1: ENCOUNTERS OVERVIEW

a. How many total encounters occurred each year?

b. For each year, what percentage of all encounters belonged to each encounter class (ambulatory, outpatient, wellness, urgent care, emergency, and inpatient)?

c. What percentage of encounters were over 24 hours versus under 24 hours?

Questions we are trying to answer

OBJECTIVE 2: COST & COVERAGE INSIGHTS

a. How many encounters had zero payer coverage, and what percentage of total encounters does this represent?

b. What are the top 10 most frequent procedures performed and the average base cost for each?

c. What are the top 10 procedures with the highest average base cost and the number of times they were performed?

d. What is the average total claim cost for encounters, broken down by payer?

Questions we are trying to answer

OBJECTIVE 3: PATIENT BEHAVIOR ANALYSIS

a. How many unique patients were admitted each quarter over time?

b. How many patients were readmitted within 30 days of a previous encounter?

c. Which patients had the most readmissions?

Data Transformation - Procedures

- Add validation check = STOP > START
- Add a new column DURATION showing the duration (in minutes) of the performed procedure

Data Transformation - Encounters

- Add a new column OVER_24_HOURS – a flag that shows if the encounter took more or less than 24 hours
- Add a new column DAYS_BETWEEN_READMISSION that shows the number of days between the previous (if available) and the current admissions date
- Add a new column READMISSION_30DAYS – a flag that shows if the readmission occurred within 30 days

Data Transformation - Payers

- Rename column Id to PAYER_ID

Data Transformation - Patients

- Mask Columns in order to protect confidential data
- Delete LAT and LON columns

Copy Data to Azure SQL Database

Create schema and subsequently, four tables

- Copy Procedures
- Copy Encounters
- Copy Payers
- Copy Patients

Reporting via Power BI